Undergraduate Dissertation

Holographic quantum error-correcting codes

A thesis submitted in fulfillment of the requirements
for the degree of Bachelor of Physics

Author:
Luis Carlos Mantilla Calderón

Supervisor:
Ph.D. Alonso Botero Mejía

Department of Physics

Bogotá, Colombia
June 30, 2021

Contents

Introduction **ii**

1 Background Information **1**
1.1 Quantum Systems . . . 1
1.1.1 Pure and Mixed States . . . 1
1.1.2 Quantum Entanglement . . . 2
1.1.3 Quantifying Entanglement . . . 3
1.2 Quantum Operations . . . 4
1.2.1 Measurements . . . 4
1.2.2 Operators on Qubits . . . 5
1.2.3 Quantum Computing: Circuits and Gates . . . 7
1.3 Tensor Networks . . . 8
1.3.1 Tensors and Tensor Networks . . . 8

2 Quantum Error Correction **11**
2.1 Quantum Error Correction in a Nutshell . . . 11
2.1.1 Quantum Error-Correcting Codes . . . 13
2.1.2 Stabilizer Codes . . . 16
2.1.3 Fault Tolerance . . . 19
2.2 AME States and Quantum Error Correction . . . 23
2.2.1 AME States . . . 23
2.2.2 Graph States . . . 25
2.2.3 Quantum Codes from AME States . . . 26

3 Holography and QEC **29**
3.1 Holography . . . 29
3.1.1 The AdS/CFT Correspondence . . . 29
3.1.2 RT Formula and AdS-Rindler Reconstruction . . . 31
3.2 Holographic Quantum Error-Correcting Codes . . . 32
3.2.1 Holographic States and Codes . . . 33
3.2.2 Recovering Properties from AdS/CFT . . . 35
3.2.3 Holographic Codes for Quantum Error Correction . . . 38
3.3 Holographic Codes for Fault Tolerance . . . 43

4 Perspectives on Holographic Codes **45**

A Classical Error Correction **47**

B Asymptotic Code Rates **49**

C Syndrome.py **55**

Introduction

Quantum Computers are very delicate devices; they suffer from decoherence when interacting with the environment. A way to suppress this effect is by using quantum error correction (QEC), a procedure in which the states that we want to manipulate on the quantum computer are encoded in the physical qubits of the computer. When these interact with the environment, the encoded state will not change drastically and could be corrected to the intended state. There are multiple classes of quantum error-correcting codes (QECC) that allow performing QEC. An important type is Stabilizer Codes [1], first introduced by D. Gottesman. These codes are popular due to their simplicity and range of applicability. Each stabilizer code can be described as the set of states invariant under a certain subgroup of the generalized Pauli group. The ideas of QEC have appeared in numerous areas of research, such as in condensed matter physics [2] as topological error-correcting codes [3], but more fascinating, they have recently made their way through the quantum gravity community [4] as they provide a new way of understanding the so called AdS/CFT correspondence, a successful realization of the holographic principle.

The holographic principle is a property that arises in quantum gravity theories, which states that a region of space can be encoded in its lower-dimensional boundary. This principle was first proposed by 't Hooft and Susskind [5] and was explicitly formulated later on by Maldacena [6] with the AdS/CFT duality. This duality is an explicit mathematical realization of the holographic principle in the string theory setup. On one side, Anti-de Sitter (AdS) spacetime is a type of spacetime with gravity characterized by having negative curvature. On the other side, Conformal Field Theory (CFT) is a type of field theory invariant under conformal maps (i.e. maps that locally preserve angles). This correspondence relates fields of a quantum gravity theory on an AdS space (bulk) of $d + 1$ dimensions with operators of a Conformal Field Theory, living on d dimensions, lying on the boundary of the AdS space.

An intriguing property of the AdS/CFT correspondence is that a bulk local observable can be reconstructed from different operators in the CFT [7], meaning that there is some redundancy in this reconstruction. Likewise, quantum error correction works by a similar principle of adding some redundancy to an encoded state such that it can be recovered from a smaller subspace of the total physical Hilbert space. The idea of interpreting the AdS/CFT correspondence as a quantum error-correcting code with the AdS fields being encoded into the CFT operators [8] motivated the creation of Holographic codes. These codes are a family of quantum error-correcting codes that were proposed for studying the AdS/CFT correspondence [9].

Almheiri et al. [4] first pointed out this idea of understanding this holographic realization as a quantum error correction code, and their proposed interpretation is being developed since then due to its significant impact on the path towards understanding quantum gravity. Recently, Pastawski et al. [8] presented an innovative construction of Holographic codes as exactly solvable toy models that realize the AdS/CFT correspondence in a Poincaré disk. The construction is based on creating a tensor network built from a particular kind of quantum states called Absolutely Maximally Entangled (AME) States [10]. These states have the property that any bipartition on two

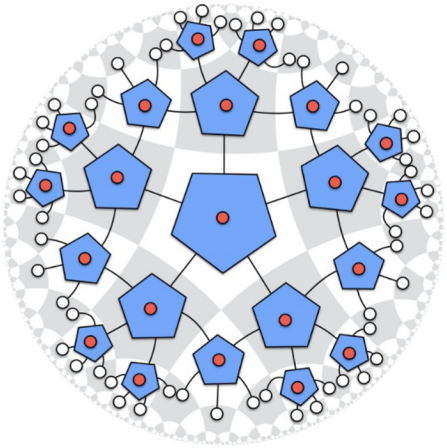


FIGURE 1: A particular HQECC known as the HaPPY code. It is constructed by placing AME(6,2) states on a pentagon tiling of the Poincaré disk. Each of the red dots represents one leg of the AME state, and they are interpreted as the input for the quantum error-correcting code. Taken from [8].

subsystems is maximally entangled. AME states are special; they do not necessarily exist for an arbitrary number of parties and local dimensions, although lots of progress has been made to understand these states. They are useful for several applications, such as teleportation of multiple states between two parties [10] and quantum secret sharing schemes [10, 11].

There are many ways to study the holographic codes introduced on [8]. Operator Algebra Quantum Error Correction (OQEC) [12] provides a clean mathematical framework to describe the action of encoding states in the bulk (AdS) into the boundary (CFT), which can be used, for example, to determine the stabilizer of particular HQECCs [13, 14]. Additionally, tensor networks [15] and graph states [16] provide tools to calculate interesting holographic properties captured by this construction, such as the Ryu-Takayanagi formula [17], the AdS-Riendler reconstruction, and the entanglement wedge reconstruction, which are some of the main features of this model.

This bachelor's thesis will provide the necessary background for the reader to understand the motivation behind holographic codes and will study the usefulness of this code family. We will focus on two particular points throughout the document:

1. Discussing why this framework helps explaining some AdS/CFT correspondence results, like the AdS-Rindler reconstruction and the RT formula.

2. Understanding if the constructed holographic codes (whose principal purpose is studying holography) are helpful, in any way, for doing quantum error correction.

To do this, we will first do a brief introduction to quantum mechanics for someone without such a background. Then, we will introduce quantum error correction, stabilizer codes, and fault tolerance. With those topics in hand, we will explain the motivation behind holographic codes and use tensor networks for constructing them. Afterwards, we calculate code properties like asymptotic code rates for some of these codes and ultimately discuss the perks and challenges of eventually using this family of codes for performing error correction in a quantum computer.

Chapter 1

Background Information

This chapter introduces some concepts from quantum mechanics that we will use throughout the next two chapters. Someone that has taken quantum mechanics 1 and 2 will likely know most of the content. Still, there might be some additional information that the reader could find interesting. For a more detailed review on these topics, I recommend references [18, 19].

1.1 Quantum Systems

1.1.1 Pure and Mixed States

In quantum mechanics, all the information that could be known about a system to describe its evolution is encoded in a **pure quantum state**, usually denoted by $|\psi\rangle$. All of the possible quantum states of a system are elements of a Hilbert Space $\mathcal{H}$, and are called "kets". Likewise, the elements of the dual space $\mathcal{H}^* = \{\langle\psi| \,|\, \langle\psi| : \mathcal{H} \longrightarrow \mathbb{C}\}$ are called "bras". As a simple case, let us consider a qubit. It is a quantum state of a two-level quantum system, in this case, the Hilbert space that describes the system is $\mathbb{C}^2$; it could be the spin of an electron, a photon's polarization, a excitation level of a trapped ion, or the current direction on a specific superconducting circuit. Two complex numbers can describe the state since there are two degrees of freedom. Having chosen a basis $\{|a\rangle, |b\rangle\}$ for $\mathbb{C}^2$, the qubit will be of the form

$$|\psi\rangle = c_a\,|a\rangle + c_b\,|b\rangle\,, \quad c_a, c_b \in \mathbb{C}.$$

Naturally, the two-level system has two physical levels, which are characterized by two orthogonal states. They are called the **computational basis** and are written as $|0\rangle$ and $|1\rangle$. Additionally, there is a restriction on the physical states, they must be normalized, i.e. $|\langle\psi|\psi\rangle|^2 = 1$. It implies for the qubit that $|c_a|^2 + |c_b|^2 = 1$. This restriction can be formalized in saying that the physical states of a particular quantum system described by a (finite) Hilbert space $\mathcal{H} \cong \mathbb{C}^n$ is the spaces of rays of $\mathcal{H}$, also known as the projective space $\mathbb{C}P^{n-1} := S^{2n+1}/U(1)$. We will denote this space of states by $\mathcal{S}(\mathcal{H})$.

Additionally, the evolution of pure quantum states is described by unitary operators, these are operators U such that $U^\dagger U = UU^\dagger = \mathbf{1}$. This is because the probability density of a state $|\psi\rangle$, defined by $|\,\langle\psi|\psi\rangle\,|^2$ should remain a probability density (adding up to one). The time evolution of $|\psi\rangle$ is governed by the Schrodinger equation:

$$i\hbar\frac{d|\psi\rangle}{dt} = H|\psi\rangle, \tag{1.1}$$

where H is the Hamiltonian of the system. Thus, if we want a quantum system to be modified in a specific way in the laboratory, we must adjust the Hamiltonian of the system to evolve it how we want.

The idea of an statistical ensemble can be extended in the quantum regime. A **mixed quantum state** is an statistical ensemble of pure quantum states $\{|\psi_i\rangle\}_i$, and it is represented by a **density matrix**

$$\rho = \sum_i p_i \, |\psi_i\rangle \, \langle\psi_i| \, ,$$

where $\sum_i p_i = 1$. We will denote the space of density matrices by $\mathcal{D}(\mathcal{H})$. Since every state, pure and mixed, can be viewed as a density matrix, we will generally use this formalism and the braket formalism for pure states interchangeably. Similarly to the pure state case, the time evolution of a mixed state ρ is driven by the von Neumann equation

$$i\hbar \frac{\partial \rho}{\partial t} = [H, \rho].$$

1.1.2 Quantum Entanglement

A key feature of quantum systems that classical systems do not exhibit is quantum entanglement, a type of correlation that is stronger than any classical correlation. To understand why entanglement is inevitable in quantum mechanics, we need to distinguish physical systems' composition. In classical mechanics, the space of states is a phase space, the collection of all generalized coordinates and momenta of the system, which mathematically is the cotangent bundle T^*Q_i of some manifold Q_i. A classical system composed of two subsystems A and B, with phase spaces T^*Q_A and T^*Q_B, respectively, will be described by the Cartesian product $T^*Q_A \times T^*Q_B$. In quantum mechanics, if a system is composed of two subsystems A and B, with Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$ respectively, the composite system will be described by states in the tensor product $\mathcal{H}_A \otimes \mathcal{H}_B$. It is clear that in the quantum mechanical case, the dimension grows exponentially in the number of systems. In contrast, in the classical case, the dimension grows linearly in the number of systems. It is in these massive Hilbert spaces of quantum mechanics where entanglement can occur.

Quantum entanglement can happen in a bipartite or a multipartite system. The former case is well understood, but the latter appears to be much more complicated. Let us give a precise definition of entanglement. For the bipartite case, meaning that the state is shared between two parties A and B, a state $|\psi\rangle$ is said to be separable if it can be written as

$$|\psi\rangle = |\phi\rangle_A \otimes \left|\phi'\right\rangle_B ,$$

otherwise, the state is **entangled**. The definition is similar for the multipartite case, meaning that the state is now shared between n parties. A state $|\psi\rangle$ is said to be separable if it can be written as

$$|\psi\rangle = |\phi\rangle_1 \otimes \cdots \otimes \left|\phi'\right\rangle_n ,$$

likewise, it is entangled if it cannot be written as a product. A canonical example of bipartite states with entanglement are the Bell states

$$\left|\phi^{\pm}\right\rangle = \frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle), \quad \left|\psi^{\pm}\right\rangle = \frac{1}{\sqrt{2}}(|01\rangle \pm |10\rangle),$$

where $|ij\rangle := |i\rangle \otimes |j\rangle$. These states are the key ingredient for quantum teleportation, quantum key distribution (QKD), and many more quantum protocols.

Entanglement is one of the main reasons why many-body systems are hard to simulate classically. Since the Hilbert space of a many-body system is exponential in the number of bodies, it is eventually unfeasible to store every matrix element of the wave function of these systems. There are many ways of getting around this problem with classical computers. For example, tensor networks (we will explain them in greater detail later in the chapter) are a tool that allows approximating entangled states with tensors of lower dimension. In any case, the best solution for accurately simulating quantum mechanics will be using quantum computers [20].

1.1.3 Quantifying Entanglement

We have discussed what entanglement is, but is there a way of "counting" entanglement? And if so, how should you do it?. A modern understanding of entanglement is thinking of it as a resource (just like money, it is a resource). If you have multiple entangled qubits available, you can perform quantum cryptography protocols and quantum teleportation protocols for a communication way more secure than a classical protected one. A way of studying entanglement is with LOCC (local operations and classical communication) operations [19]. These are all the physical operations that two parties can perform by sharing classical information (calling each other by cellphone with unlimited text and talk plans) and doing local operations in their laboratories. The set of LOCC operations is used since entanglement can only be created locally and cannot suddenly happen between far-away systems if they were never physically together in the past. Vidal showed that a correct way of quantifying the "amount" of entanglement stored between parts of a quantum system is using entanglement monotones [21]. These are real-valued functions over the quantum states that cannot increase under LOCC operations.

For the bipartite pure case, a special entanglement monotone is the **von Neumann Entropy**, defined as

$$S(\rho) = -\sum_{i=1}^{N} \lambda_i \log_2 \lambda_i, \tag{1.2}$$

where $\vec{\lambda} = (\lambda_1, \ldots, \lambda_n)$ is the vector of the singular value decomposition coefficients (the generalization of eigenvalues) of ρ, also known as the Schmidt vector. The von Neumann entropy is a specific case of a class of functions of the Schmidt vector that classify entanglement, called Schur-concave functions. To understand this better, let us remind what majorization is:

Definition 1.1.1. Let $a, b \in \mathbb{R}^n$ be two real-valued vectors. Denote $a^{\downarrow}$ to the vector of a with the components reorganized in descending order ($a_i^{\downarrow} \leq a_j^{\downarrow}$ if $i \leq j$). Then, a is majorized by b ($a \prec b$) if

$$\sum_{i=1}^{k} a_i^{\downarrow} \leq \sum_{i=1}^{k} b_i^{\downarrow}, \quad \forall k \in \{1, \ldots, n\}.$$

The **Schur-concave** functions are real-valued functions that invert the majorization relation between two vectors. If f is Schur-concave and $a \prec b$, then $f(a) \geq f(b)$. Nielsen showed [22] that these functions play a crucial role in entanglement, particularly he proved the following theorem:

Theorem 1.1.1. *In the pure bipartite scenario, a quantum state $|\psi\rangle$ can be transformed to $|\phi\rangle$ by means of LOCC operations if and only if the Schmidt vector of ϕ is*

majorized by the Schmidt vector of ψ

$$|\psi\rangle \overset{\text{LOCC}}{\longrightarrow} |\phi\rangle \iff \vec{\lambda}_\psi \prec \vec{\lambda}_\phi.$$

Therefore, candidates for entanglement monotones (functions that quantify entanglement) are Schur-concave functions E acting on the Schmidt vectors of different pure quantum states. Thus, for two states $|\psi\rangle$ and $|\phi\rangle$,

$$\vec{\lambda}_\psi \prec \vec{\lambda}_\phi \quad \implies \quad E(\vec{\lambda}_\psi) \geq E(\vec{\lambda}_\phi).$$

To extend a entanglement monotone E to mixed states $\rho \in \mathcal{D}(\mathcal{H})$ it is common to consider the convex roof function

$$\tilde{E}(\rho) = \inf \sum_{i \in I} p_i E(\psi_i),$$

where the infimum is taken over $\left\{ \{(p_i, \psi_i)\}_{i \in I} \mid \rho = \sum_{i \in I} p_i |\psi_i\rangle \langle\psi_i| , \sum_{i \in I} p_i = 1 \right\}$ all the possible ensembles that reproduce ρ.

In the multipartite case, quantifying genuine multipartite entanglement is much more challenging and is still an active area of research. We will not address measures like the 3-tangle and the Schmidt rank for multipartite systems in this thesis, but if the reader wants to deep further into entanglement theory, a good place to start is the review by Horodecki et al. [19].

1.2 Quantum Operations

1.2.1 Measurements

In classical mechanics, measurements are real-valued smooth functions over phase space. An example would be having a ruler and measuring some distance on a system. Meanwhile, in quantum mechanics, measurements drastically affect the system and cannot be modelled by smooth functions. It is a postulate of quantum mechanics [18] that quantum measurements are described by a set of operators $\{M_m\}$ acting over the Hilbert space $\mathcal{H}$. The outcomes of such a measurement (in other words, the values that can take your "quantum ruler") are indexed by m. These operators must satisfy the completeness equation

$$\sum_m M_m^\dagger M_m = I,$$

meaning that if a state is measured there will always be an outcome. Additionally, the probability $\mathbb{P}(m)$ of obtaining m from measuring a pure state $|\psi\rangle$ is

$$\mathbb{P}(m) = \left\langle \psi \middle| M_m^\dagger M_m \middle| \psi \right\rangle,$$

and after the measurement, the state $|\psi\rangle$ is updated to the new state

$$\left|\psi'\right\rangle = \frac{M_m|\psi\rangle}{\sqrt{\left\langle \psi \middle| M_m^\dagger M_m \middle| \psi \right\rangle}}.$$

There are multiple types of quantum measurements. The most common ones are **projective measurements** and **POVM measurements** (positive operator-valued measurements). The former refers to a set of operators that are orthogonal,

meaning that $P_m P_k = \delta_{mk} P_m$. They commonly appear as the projections onto the eigenspaces of Hermitian operators. The latter refers to a more general set of operators $\{E_m\}_m$ that satisfy the completeness equation $\sum_m E_m = I$, and the outcome m has probability $\langle\psi| E_m |\psi\rangle$ of being measured. We can realize every POVM as a projective measurement on a larger Hilbert space (this is known as Naimark's dilation theorem). However, it is beneficial for an experimental treatment to use POVMs since quantum systems interact with the environment. Unless you have control over the environment (which generally is not the case), you will now have access to the complete quantum state and will need to describe measurements with POVMs.

Similarly, we can describe measurements in the density matrix language. For a state $\rho = \sum_j p_j |\psi_j\rangle\langle\psi_j|$, the expected value of a measurement N is

$$\langle N \rangle = \sum_j p_j \langle\psi_j| N |\psi_j\rangle = \mathrm{tr}(\rho N).$$

We can now decompose N as $N = \sum_k n_k N_k$ where N_k is a projection onto the subspace of $\mathcal{H}$ with measurement outcome (eigenvalue) k. This allows us to determine the updated state ρ' obtained after measuring ρ with N. If the outcome of N is k, the state will collapse to $\rho_k = \frac{N_k \rho N_k}{p_k}$ where $p_k = \mathrm{tr}(\rho N_k)$ is the probability of measuring k. Finally, in the case where we do not know the outcome value of N, the post-measurement state ρ' will be a superposition of all possible post-measurement states ρ_k:

$$\rho' = \sum_k p_k \rho_k = \sum_k (\mathrm{tr}(\rho N_k)) \left(\frac{N_k \rho N_k}{\mathrm{tr}(\rho N_k)} \right) = \sum_k N_k \rho N_k.$$

Measurements are essential for implementing quantum error correction on a quantum computer (we will talk about this in the next chapter). For example, in gate-based quantum computing [18], the standard measurements used for quantum error correction are projective measurements of ancillas coupled to the system. If done incorrectly, we can significantly alter the result of a computation.

1.2.2 Operators on Qubits

To control a quantum system, we must make it evolve in the desired manner. We mentioned that the temporal evolution of quantum systems is unitary. Still, generally, we cannot describe these evolution operators as unitary matrices since the interaction between the environment and the controlled system will prevent us from accessing all the relevant degrees of freedom of the composite system. This motivates the definition of CPTP maps, which are a generalization of unitary operators:

Definition 1.2.1. A completely positive trace preserving map (**CPTP map**) is a linear operator $\mathcal{O} : \mathcal{H}_1 \to \mathcal{H}_2$ such that it is

1. **Positive:** For every non-negative $\rho \in \mathcal{D}(\mathcal{H}_1)$, $\rho \geq 0 \implies \mathcal{O}(\rho) \geq 0$.
2. **Completely positive:** If R is a reference system (also called *ancilla*), for a density operator $\sigma \in \mathcal{D}(\mathcal{H}_R \otimes \mathcal{H}_1)$,
$$\mathrm{tr}_R(\sigma) = \rho \text{ and } \sigma \geq 0 \implies (I_R \otimes \mathcal{O})(\sigma) \geq 0.$$
3. **Trace preserving:** For every $\rho \in \mathcal{D}(\mathcal{H}_1)$, $\mathrm{tr}(\rho) = \mathrm{tr}(\mathcal{O}(\rho))$.

This operation accounts for the most general transformation that a quantum system can undergo. The names CPTP and **quantum channel** are used interchangeably in the literature. These maps admit a decomposition into sums of matrices, known as the **Kraus representation** stated in the following theorem:

Theorem 1.2.1. *Let $\mathcal{O} : \mathcal{H}_1 \longrightarrow \mathcal{H}_2$ be a CPTP map, with $\dim(\mathcal{H}_1) = n$ and $\dim(\mathcal{H}_2) = m$. Then, there exist a non-unique set of matrices $\{A_k\}_{k\in\{1,\dots,nm\}}$ such that*

$$\mathcal{O}(\rho) = \sum_{k=1}^{nm} A_k \rho A_k^\dagger,$$

with $\sum_k A_k^\dagger A_k = I$. This decomposition is known as the Kraus representation of $\mathcal{O}$.

As you might be thinking, temporal evolution in quantum mechanics is always unitary, and generally, CPTP maps are not. What is occurring is that the controlled system has coupled to the environment, and since you only have access to the controlled one, tracing out the environment will give rise to a CPTP map. This fact is the realization of the Stinespring dilation theorem [23], which informally states that every CPTP map can be obtained by first tensoring the controlled system with an ancilla, then applying a unitary map on the composite system, and lastly, tracing out the ancilla.

Having this in mind, we can now focus on particular operators relevant for quantum error correction. In qubit systems, some of the standard gates (unitary matrices) are the Pauli matrices. The Pauli matrix X, also known as the NOT or bit-flip gate, flips the states $|0\rangle$ and $|1\rangle$. The Pauli matrix Z, called the phase gate, changes the phase of the state $|1\rangle$ by π. The Pauli matrix Y is the product of Z and X gates. Additionally, the Hadamard gate H creates superposed states, the rotation R_θ introduces a phase into the qubit, and the CNOT allows entangling two qubits. The matrix representation in the computational basis for such operators is the following:

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad Y = iXZ = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix},$$

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad R_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{2i\theta} \end{pmatrix}, \quad \text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

We can generalize the X and Z gates to qudits $\mathcal{S}(\mathbb{C}^d)$. These gates will act on the computation basis $\{|0\rangle, \dots, |j\rangle, \dots, |d-1\rangle\}$ as

$$\begin{aligned} Z|j\rangle &= \omega^j |j\rangle, \quad \omega = e^{2\pi i/d} \\ X|j\rangle &= |j +_d 1\rangle. \end{aligned}$$

Thus, in the computational basis, $Z = \text{diag}(1, \omega, \omega^2, \dots, \omega^{d-1})$ and X is be the permutation matrix that sends $|j\rangle$ to $|j +_d 1\rangle$. These gates are standard operations used in quantum algorithms, and we will be using them for the next chapter in quantum error correction. Mainly, the group of operators composed of products of the X and Z gates helps to study general qubit errors. This group is called the Pauli group.

Definition 1.2.2. The **Pauli group** of a qudit of dimension d is defined by

$$\mathcal{P} = \{\omega^j X^k Z^l \mid j, k, l \in \mathbb{Z}_d\}.$$

Moreover, the generalized Pauli group on n-qudits $\mathcal{P}_n$ consists of the n-fold tensor product of single qudit Pauli groups:

$$\mathcal{P}_n = \mathcal{P}^{\otimes n} = \mathcal{P} \otimes \cdots \otimes \mathcal{P}.$$

For the case of qubits, the Pauli group consists of 16 operators, which $\{I, X, Y, Z\}$ generates. This group is valuable since I, X, Y, and Z are a basis of $U(2)$, the set of all unitary matrices acting on a qubit. This means that we can decompose any unitary action as a linear combination of the Pauli matrices. Furthermore, an important relationship of the d-dimensional Pauli group is the commutation between elements of this group. For $a, b \in \{0, 1, \ldots, d-1\}$, and $\omega = e^{2\pi i/d}$, we have that

$$\left(X^a Z^b\right)\left(X^{a'} Z^{b'}\right) = \omega^{ba'-ab'}\left(X^{a'} Z^{b'}\right)\left(X^a Z^b\right),$$

where $ba' - ab'$ is the symplectic product of $\mathbb{R}^2$ between (a, b) and (a', b'). In Appendix C we discuss how to classically do calculations with elements of the generalized Pauli group without the expenses of storing exponentially large matrices.

1.2.3 Quantum Computing: Circuits and Gates

Quantum error correction emerged to solve one problem: prevent errors from damaging quantum computations. But what is quantum computing? And why do companies like Microsoft [24] and PsiQuantum [25] are working on building one of these computers? Quantum computing is a computation model that promises algorithmic speed advantages for many algorithms in classical models of computation. Typical examples are Shor's algorithm and Grover's algorithm, which give speed advantages for factoring numbers and finding elements on databases compared to classical programs, respectively. We can implement these algorithms in any of the various universal quantum computation models, like adiabatic quantum computation [26], measurement-based quantum computation [27], or the quantum circuit model [18]. All of these computation models are equivalent because each of them can efficiently simulate the others. For this thesis, the quantum circuit model will provide a clearer picture for most definitions. Analogous to the classical circuit model, this model consists of a sequence of gates applied to qubit registers and a set of measurements of these registers. Contrary to the classical circuit model, gates must be reversible operations due to the unitary nature of quantum mechanics.

To build a quantum computer that can run any quantum algorithm, it must perform any operation on the qubits in an efficient manner. This task can be achieved using a finite set of **universal gates**, meaning that any other gate can be approximated with arbitrary precision with gates from the chosen set. An example of a universal sets of gates are the CNOT, Hadamard, $R_{\pi/4}$ and $R_{\pi/8}$ gates. Formally, a collection of gates A is said to be universal if the subgroup generated by A is dense in

$$SU(2^n) := \{M \in U(2^n) : \det(M) = 1\}.$$

The **Solovay-Kitaev theorem** assures that with a universal set of gates S, we can approximate any $SU(2^n)$ gate with a precision of ϵ using a sequence $g_1 \ldots g_n$ of gates $g_i \in S$ of a length $n = \mathcal{O}(\log^c(\epsilon^{-1}))$. As we will see in the upcoming chapter,

implementing a universal set of gates and achieving error robustness for quantum computation can sometimes be challenging.

1.3 Tensor Networks

To finish this chapter of background information, we will introduce tensor networks. In a few words, these are a tool for approximating wave functions of many-body systems. We will use them in the third chapter to understand and construct holographic codes. For a detailed and complete review of tensor networks, I suggest Biamonte's lecture notes [15].

1.3.1 Tensors and Tensor Networks

(A) (B)

FIGURE 1.1: Tensor diagram notation of (A) a tensor and of (B) a tensor network. The tensor in (A) is an order 6-tensor, and when each leg is identified with an index, this circle outputs a number. The tensor network in (B) represents the order 5 tensor that results from the contraction between some legs of the tensors A,B,C, and D.

A finite-dimensional tensor is a multilinear map that, after a choice of basis, can be thought of as a multidimensional array. Graphically, we can draw it as a circle with legs, where the circle outputs a number when each leg is identified with an index; this notation is called the tensor diagram notation. In Figure 1.1a, a 6 order tensor T is pictured as a circle with 6 legs. The array element T_{ijklmn} is the entry $ijklmn$ of T, and in the diagram notation, it corresponds to identifying each leg with its corresponding index. Certainly, tensors are a powerful tool since they are the mathematical concept that formalizes quantum operations and quantum states. We can generalize the notion of tensors to tensor networks, where now we contract legs of different tensors (Figure 1.1b). The tensor that remains after the contraction of a leg of two different tensors is the sum over all possible indexes of the contracted leg. For example, the contraction of T_{ij} with R_{kj} along the second legs is $\sum_j T_{ij}R_{kj}$. From this definition, it is clear that both contracted legs must "belong" to isomorphic Hilbert spaces (which for finite dimension means that both spaces have the same dimension). In other words, the number of indices that label the entries of two contracted legs must be the same.

We can now interpret tensors as quantum states. For qudit systems, the wave functions are some weighted superposition of a chosen basis. The weights of the sum are a multidimensional array of numbers, in other words, a tensor. So, from a given

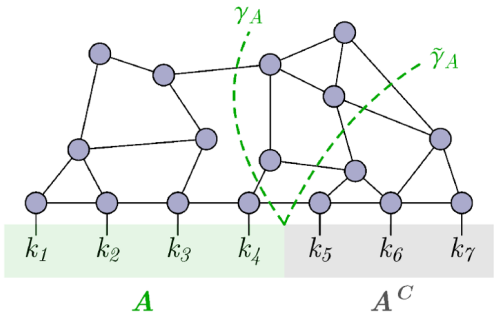


FIGURE 1.2: Two sub-regions of a tensor network separated by two cuts, γ_A and $\tilde{\gamma}_A$. Taken from [28].

a tensor T with n legs and each leg j of dimension d_j, we can build a quantum state of n qudits, each of dimension d_j as

$$|\psi\rangle = \sum_{\substack{1\leq k_1\leq d_1 \\ \vdots \\ 1\leq k_n\leq d_n}} T_{k_1\cdots k_n} |k_1\rangle \cdots |k_n\rangle .$$

Thus, the idea of approximating many-body systems with tensor networks is to model the huge dimensional tensor T (generally, classically intractable) that describes the many-body state $|\psi\rangle$ as a contraction of smaller (classically tractable) tensors Q_i. In terms of quantum states, we are restricting ourselves to a smaller subset of the Hilbert space.

For states described by tensor networks, we can calculate values such as the entanglement entropy between two disjoint regions, and this will be something we will be doing in Chapter 3 for particular states. Suppose we have the tensor network of Figure 1.2, and we want to calculate the entanglement entropy between region A and region A^c. We can interpret the tensor network as a linear map $T_A : A \to A^c$. Then, with the reduced density matrix $\rho_A = T_A^* T_A$, we can find the entanglement entropy using the definition:

$$S(\rho_A) = -\operatorname{tr}(\rho_A \log \rho_A) = S(\rho_{A^c}).$$

For arbitrary tensors, we can bound this entropy by noting that the *rank* of the linear map T_A satisfies the following bound:

$$\operatorname{rank}(T_A) \leq \min \left\{ \prod_{b\in\gamma_A} d_b \mid \gamma_A \text{ is a cut separating } A \text{ from } A^c \right\},$$

where d_b is the dimension of a bond b that γ_A cuts. So, using the definition of the entanglement entropy of ρ_A and the previous bound, we obtain that

$$S(\rho_A) \leq \log\left(\operatorname{rank}(\rho_A)\right) \leq \min \left\{ \sum_{b\in\gamma_A} \log d_b \mid \gamma_A \text{ is a cut separating } A \text{ from } A^c \right\}.$$

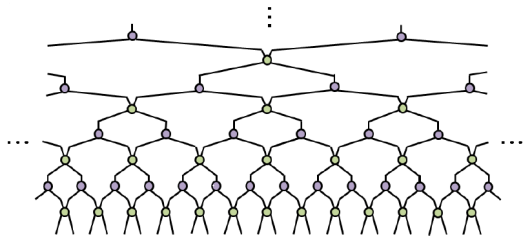


FIGURE 1.3: A scale-invariant MERA consisting of disentanglers (green tensors) and isometries (purple tensors). Taken from [30].

To correctly approximate a quantum state $|\psi\rangle$, we need to choose a tensor network architecture that allows achieving high fidelity and recovering the expected entanglement properties of the state $|\psi\rangle$. An example to consider is the ground states of local and gapped 1-D Hamiltonians. These states follow an area law [29], meaning that the entanglement entropy of a region A is proportional to its boundary $|\partial A|$. These states' wave function can be approximated by what is called a matrix product state, a contraction of a train of tensors that recovers the observed entanglement properties.

There are different tensor network structures for other wave functions' classes that reproduce the wave functions' entanglement properties. The most common ones are matrix product states (MPS) [15], projected entangled pair states (PEPS) [15], and multi-scale entanglement renormalization ansatz (MERA) [30]. A relevant tensor network structure that will be relevant for this thesis is MERA (Figure 1.3). This tensor network approximates states with long-range entanglement as a contraction of two types of tensors: isometries and disentanglers.

Throughout the development of tensor networks, the multi-scale entanglement renormalization ansatz (MERA) transformed the study of long-range entanglement, which is complicated using MPS or PEPS. It introduced an architecture (Figure 1.3) that allowed approximating certain states that did not strictly follow an area law but had a logarithmic correction on the entanglement entropy. Swingel pointed out that the MERA for the ground state of a spin chain can be interpreted as a discrete realization of the AdS/CFT correspondence (this is discussed in Chapter 3). Ultimately, the development of this tool allows numerical simulations of many-body quantum states that play a role in AdS/CFT, condensed matter physics, and chemistry, which by brute-force computations would require an excessive amount of computational resources.

Chapter 2

Quantum Error Correction

This chapter aims to introduce the theory of quantum error correction and some significant results. We discuss quantum codes, stabilizer codes, fault tolerance, and a particular family of codes obtained from AME states that will be used to build holographic codes in the third chapter. I will follow Daniel Gottesman's lectures [31], Andrew Steane's [32] and John Preskill's [33] lecture notes on quantum error correction, which I highly recommend for someone entering the field and wanting to learn about the topic. Other great sources for learning quantum error correction are the books by Nielsen [18], and Lidar [12].

2.1 Quantum Error Correction in a Nutshell

Let us first explain what the essence of error correction is from a classical point of view. Suppose Alice wants to send a binary message to Bob through a noisy channel. The noisy channel could be a physical phone wire or a hard drive sending information from the past (Alice) to the future (Bob). If, with probability $p < 0.5$, a transmitted bit flips, the noisy channel could distort the message that Alice is sending, and Bob will receive it altered. One way of reducing Bob's chances of receiving a wrong message could be for Alice to repeat the bits multiple times. For instance, Alice sends 000 to represent a 0 and 111 to represent a 1. This repetition encoding is a map from $\{0,1\}$ to $\{0,1\}^3$ whose action is $0 \to 000$ and $1 \to 111$. When Bob receives the string of bits, he will decode it by taking the most common bit out of every three consecutive received bits. For example, if he gets 110010111100101000, he will decode it to 101010. If Alice and Bob use this protocol, the probability of error per bit is reduced from p to $3p^2(1-p)+p^3$. If we want to minimize this error probability more, one way of doing so is by repeating a bit more than three times, but this approach is not the smartest one. The code rate for a code that encodes k bits in N bits is defined by $R = \frac{k}{N}$, and for the repetition code, this rate is $1/N$. To arbitrarily reduce the error probability using the repetition code, we need to sent $N \to \infty$, achieving an asymptotic code rate of 0 (extremely inefficient). For a code to be considered acceptable, this code rate should at least be positive. But to precisely know if the code is good or bad, we must recall Shannon's source coding theorem [34].

Theorem 2.1.1. *We can compress N independent and identically distributed random variables $\{X_1, \ldots, X_N\}$, each with entropy $H(X_1)$, into more than $NH(X_1)$ bits with arbitrary protection as $N \to \infty$.*

Then, the closer the code rate is to the Shannon entropy of the message, the better the code is. Examples of more efficient error-correcting codes than the repetition code are the Reed-Solomon codes the Hamming codes, each of which has a positive code rate. A more detailed introduction to classical linear codes is given in Appendix A.

In the quantum case, if Alice wants to send a qubit to Bob, like a polarized photon, though a noisy optical fibre (modelled as a CPTP map), we cannot use the repetition protocol mentioned above. The no-cloning theorem prohibits implementing a repetition code. It states the following:

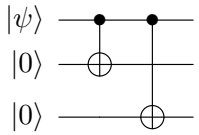
Theorem 2.1.2. *There does not exist a quantum operator that clones arbitrary states.*

Proof. By contradiction, assume A is the cloning operator, then if $|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle$ with $c_0 c_1 \neq 0$, we have the following contradiction

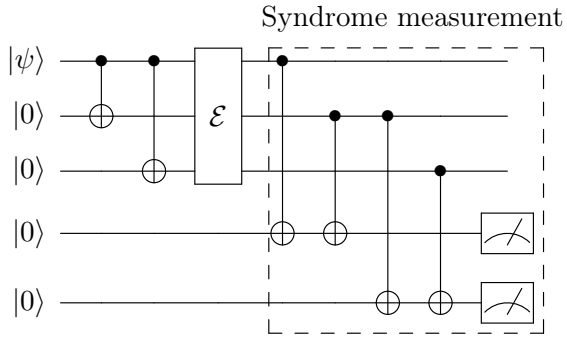
$$\begin{aligned} c_0^2 |00\rangle + c_1^2 |11\rangle + c_0 c_1 (|01\rangle + |10\rangle) &= |\psi\rangle \otimes |\psi\rangle \\ &= A(|\psi\rangle \otimes |0\rangle) \\ &= A(c_0 |0\rangle \otimes |0\rangle + c_1 |1\rangle \otimes |0\rangle) \\ &= c_0^2 |00\rangle + c_1^2 |11\rangle . \end{aligned}$$

□

This restriction is a problem because we cannot directly use the error correction machinery developed in classical codes for quantum errors. We need to find a way around it. The key idea from classical error correction was introducing some redundancy on the bits. We can follow this approach and redundantly encode qubits in an entangled state of more physical qubits. A significant benefit of encoding on an entangled global state is that local errors will not severely impact the encoded information. Following the idea of redundantly encoding instead of copying a state leads us to an analogous example to the classical repetition code. To protect a qubit $|\psi\rangle$ from a noisy bit-flip channel $\mathcal{E}(\rho) = (1-p)\rho + pX\rho X$, we can encode the state $|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle \mapsto c_0 |000\rangle + c_1 |111\rangle$ by using CNOT gates over ancillas (note that this is not cloning because $c_0 |000\rangle + c_1 |111\rangle \neq |\psi\rangle \otimes |\psi\rangle \otimes |\psi\rangle$).



Another problem that we now face is that measuring any of the three qubits would collapse the state, so identifying if some error has happened or not is much more subtle. The way of doing this is by measuring the so-called error syndromes with additional ancillas. Error syndromes are a parity check of certain operators that determine the type of error that has occurred.



For example, in the quantum "repetition" code that we mentioned previously, the idea with a syndrome measurement is to check whether some bit flip has happened

in any qubit. As the above circuit shows, the first ancilla determines the difference between the first and second qubits, while the second ancilla determines the difference between the second and third qubits. Thus, depending on the measurement outcome, we will decide whether we need to fix the state by applying an X gate to a specific qubit or not. Suppose that there is a bit flip on the second qubit, so the encoded state is now $\psi = c_1 |010\rangle + c_2 |101\rangle$. The ancilla measurement outcome will be (1,1), indicating that the first and second qubits differ, and the second and third qubits differ. For this to happen, either the first and third qubits suffered a bit flip, or the second qubit did. Since the latter has a higher probability, we should apply an X gate on the second qubit to fix the error. This protocol will effectively change the channel $\mathcal{E}$ to $\mathcal{E}'(\rho) = \left((1-p)^3 + 3p(1-p)^2\right)\rho + (3p^2(1-p) + p^3)X\rho X$, reducing the chances of error.

The downside of this encoding is that it only helps to fix bit-flip errors. But if our quantum channel has more complicated errors than a single bit flip, it will not work. To achieve a robust protocol against more general errors, we must use a better error-correcting code. Despite seaming a daunting task, there is a famous example of a code that can fix every single-qubit error, and it is the nine-qubit Shor code [12]. It can fix any Pauli error (X, Y and Z), and since Pauli matrices are a basis for the 2×2 unitary matrices, it will fix any single-qubit unitary error. We will look at this code in better detail when studying stabilizer codes.

2.1.1 Quantum Error-Correcting Codes

Quantum states are incredibly fragile, and many possible errors can affect them. There is a continuum of possible errors since the slightest phase rotation of a qubit affects the state. The problem we face is like error-correcting on analog computing, and at first sight, it seems impossible. To overcome this, as Steane mentions, we can solve it by performing a "digitization of noise" by measuring some operators on the noisy state. The way the "digitization of noise" is done is by simply doing a syndrome measurement. This will collapse a superposition of continuous errors to a discrete set of errors affecting the system. To understand it better, let us go back to the repetition code we mentioned before. If the state were to have an x-axis rotation error on the second qubit

$$R_\phi = \begin{pmatrix} \cos\frac{\phi}{2} & -i\sin\frac{\phi}{2} \\ -i\sin\frac{\phi}{2} & \cos\frac{\phi}{2} \end{pmatrix} = \cos\frac{\phi}{2} I - i\sin\frac{\phi}{2} X,$$

by measuring the error syndrome as before we would collapse the rotation error into either a state without errors with probability $\left|\cos\frac{\phi}{2}\right|^2$ or a state with a bit-flip error on the second qubit with probability $\left|\sin\frac{\phi}{2}\right|^2$. For both cases, we know how to fix the error (if any). In some sense, we are attacking "bad" entanglement (errors produced by the interaction of the state with the environment) with more entanglement (coupling the ancillas and the state). Generally, since every error on an n-qubit state can be expressed as linear combinations of Pauli group elements, we can collapse an arbitrary unitary error to a Pauli error. The upcoming definitions will be helpful to classify the different types of errors that a quantum system can suffer.

Definition 2.1.1. The **weight** of an operator $\mathcal{O} = E_1 \otimes E_2 \otimes \cdots \otimes E_n$ is the number of non-identity factors E_i.

For example, for the quantum repetition code in a bit-flip error channel (discussed above), the set of possible errors contains a one weight 0 operator $I \otimes I \otimes I$; three

weight 1 operators $X\otimes I\otimes I$, $I\otimes X\otimes I$, $I\otimes I\otimes X$; three weight 2 operators $X\otimes X\otimes I$, $X\otimes I\otimes X$, $I\otimes X\otimes X$; and one weight 3 operator $X\otimes X\otimes X$. This is basically the Kraus decomposition of the quantum channel, and it allows classifying the different types of errors.

Definition 2.1.2. A CPTP map on n-qubits that admits a Kraus decomposition $\{A_j\}_{j\in I}$ such that every operator A_j has a weight less than k is known as a **k-qubit error**.

Some commonly used qubit error models that model idealized noisy quantum channels (real-life quantum errors are much worse) are the depolarizing channel

$$\rho \mapsto (1-p)\rho + \frac{p}{3}\left(X\rho X + Y\rho Y + Z\rho Z\right),$$

the dephasing channel

$$\rho \mapsto \left(1-\frac{p}{2}\right)\rho + \frac{p}{2}Z\rho Z,$$

and the amplitude-damping channel

$$\rho \mapsto \begin{pmatrix}1 & 0\\ 0 & \sqrt{1-p}\end{pmatrix}\rho\begin{pmatrix}1 & 0\\ 0 & \sqrt{1-p}\end{pmatrix} + \begin{pmatrix}0 & \sqrt{p}\\ 0 & 0\end{pmatrix}\rho\begin{pmatrix}0 & \sqrt{p}\\ 0 & 0\end{pmatrix}.$$

To decrease the probability of error of these and other quantum channels, as we did with the repetition code, we need a protocol that adds redundancy to the state and is designed to prevent errors from the chosen error model.

Definition 2.1.3. A **quantum error-correcting code** is a partial isometry (called an encoding)

$$\mathcal{O} : \mathcal{H}^k \longrightarrow \mathcal{H}^d,$$

together with a set of errors $\mathcal{E}$ such that there exist a recovery channel $\mathcal{R}$ that corrects all the errors in $\mathcal{E}$. This means that $\forall\,|\psi\rangle \in \mathcal{H}^k$ and $\forall E \in \mathcal{E}$, the following equality holds

$$\mathcal{R}(E\mathcal{O}\,|\psi\rangle\langle\psi|\,\mathcal{O}^\dagger E^\dagger) = c\,|\psi\rangle\langle\psi|, \quad c\in\mathbb{C}\setminus\{0\}.$$

The spaces $\mathcal{H}^k$ and $\mathcal{H}^d$ are called the logical and physical space, respectively. Also, the subspace $\mathcal{C} := \mathrm{im}(\mathcal{O}) \subseteq \mathcal{H}^d$ is known as the code subspace.

The question now is, how can we be sure that some encoding will be able to correct some set of errors $\mathcal{E}$. To do this, the effect of the errors on the codespace must be distinguishable. The only way this is possible is if for every pair of errors $E_i, E_j \in \mathcal{E}$, the subspaces $E_i\mathcal{C}$ and $E_j\mathcal{C}$ are mutually orthogonal. Formally, the necessary and sufficient condition that a quantum error-correcting code $\mathcal{O}$ with code subspace $\mathcal{C}$ must satisfy to correct a set of errors $\mathcal{E}$ is known as the **Knill-Laflamme** condition. It states that any pair of codewords from an orthonormal basis of $\mathcal{C}$ which is affected by different errors of $\mathcal{E}$ should still be orthogonal. If $\{|\psi_k\rangle\}_k$ is an orthonormal basis of $\mathcal{C}$, then

$$\langle\psi_i|\,E_a^\dagger E_b\,|\psi_j\rangle = C_{ab}\delta_{ij}, \quad \forall E_a, E_b \in \mathcal{E},$$

where C_{ab} is a Hermitian matrix. Another restatement of this same condition in terms of projectors is that if $\Pi_{\mathcal{C}}$ is the projector onto the code subspace $\mathcal{C}$, then for every $E_a, E_b \in \mathcal{E}$ we must have that

$$\Pi_{\mathcal{C}} E_a^\dagger E_b \Pi_{\mathcal{C}} = C_{ab}\Pi_{\mathcal{C}}.$$

If an encoding $\mathcal{C}$ satisfies the Knill-Laflamme condition for a set of errors $\mathcal{E}$, then a recovery channel $\mathcal{R}$ is guaranteed to exist.

Theorem 2.1.3. *If a quantum error-correcting code $\mathcal{O}$ corrects two errors E_1 & E_2, then it corrects any linear combination of both errors $\lambda_1 E_1 + \lambda_2 E_2$, $0 \neq \lambda_i \in \mathbb{C}$.*

Proof. If $\mathcal{O}$ can correct E_1 & E_2, then, there exists a quantum channel $\mathcal{R}$ with Kraus operators $\{R_k\}_k$ such that

$$\mathcal{R}\left(E_i \rho E_j^\dagger\right) = \sum_k R_k \left(E_i \rho E_j^\dagger\right) R_k^\dagger = \delta_{ij} c_i \rho, \quad c_i \in \mathbb{C} \setminus \{0\}, \quad i \in \{1, 2\}$$

for all $\rho \in \mathcal{H}_L$. So, for $\tilde{E} = \lambda_1 E_1 + \lambda_2 E_2$ we have that

$$\begin{aligned}
\mathcal{R}\left(\tilde{E}\rho\tilde{E}^\dagger\right) &= \sum_k R_k \left(\lambda_1 E_1 + \lambda_2 E_2\right) \rho \left(\lambda_1 E_1 + \lambda_2 E_2\right)^\dagger R_k^\dagger \\
&= \left(c_1 + c_2\right) \rho + \sum_k R_k \left(\lambda_1 E_1 \rho \lambda_2^\dagger E_2^\dagger\right) R_k^\dagger + \sum_k R_k \left(\lambda_2 E_2 \rho \lambda_1^\dagger E_1^\dagger\right) R_k^\dagger \\
&= \left(c_1 + c_2\right) \rho + \lambda_1 \lambda_2^\dagger \sum_k R_k \left(E_1 \rho E_2^\dagger\right) R_k^\dagger + \lambda_2 \lambda_1^\dagger \sum_k R_k \left(E_2 \rho E_1^\dagger\right) R_k^\dagger.
\end{aligned}$$

Since the subspaces $E_1\mathcal{C}$ and $E_2\mathcal{C}$ are orthogonal, $\mathcal{R}(E_1 \rho E_2) = \delta_{12} c_1 \rho = 0$. Then,

$$\mathcal{R}\left(\tilde{E}\rho\tilde{E}^\dagger\right) = (c_1 + c_2)\rho.$$

□

Here is where the idea of *digitizing* quantum noise comes into play. All the possible errors on qubits (infinitely many) can be expressed as linear combinations of Pauli operators (finitely many). So, a quantum code that can correct any k-weight Pauli error will be able to correct every possible k-weight error affecting the system. Like in classical codes, we can define the notions of *distance* and *code rate* for quantum codes, which are "measures" of how good and efficient a code is for correcting arbitrary errors.

Definition 2.1.4. The **distance** d of a quantum code encoding k qubits in n qubits is the minimum weight of a Pauli operator P_ℓ such that

$$\langle \psi_i | P_\ell | \psi_j \rangle \neq C_{\ell\ell} \delta_{ij}.$$

This code is referred as a $((n, k, d))$ quantum code (for qudits of dimension q, a subscript is added $((n, k, d))_q$).

Definition 2.1.5. The **code rate** R of a $((n, k, d))$ quantum code is the ratio between the encoded qubits and the physical qubits

$$R = \frac{k}{n}.$$

Note that the Knill-Laflamme condition implies that for a code to correct t unlocated Pauli errors, it must have a distance $d \geq 2t + 1$. Additionally, if the locations of the t errors are known, a $d = t + 1$ distance quantum code will be able to correct the errors.

As in classical coding theory, there are bounds on the distances and code rates that general quantum codes satisfy:

Theorem 2.1.4. *A quantum error-correcting code* $((n,k,d))_q$ *satisfies the following inequalities [35]*

- *Quantum Singleton bound*
$$k \leq n - 2d + 2$$
- *Quantum Hamming bound*
$$\sum_{i=0}^{\lfloor (d-1)/2 \rfloor} \left(\begin{array}{c} n \\ i \end{array} \right) \left(q^2 - 1\right)^i \leq q^{n-k}$$

A quantum code is called **perfect** if it saturates the quantum Hamming bound. These bounds allow us to deny the existence of quantum codes with values n, k, d that do not satisfy them.

2.1.2 Stabilizer Codes

A particular type of quantum error-correcting codes are *stabilizer codes*. The idea behind these codes is to encode quantum states in a linear subspace of the physical Hilbert space, which itself is invariant under a particular set of operators. Specifically, a stabilizer code is defined in the following way:

Definition 2.1.6. Let $\mathcal{S} \subseteq \mathcal{P}_n$ be an abelian subgroup of the generalized Pauli group such that $\omega I \notin \mathcal{S}$. The subspace

$$T(\mathcal{S}) := \{|\psi\rangle \in \mathcal{H}_p \mid S\,|\psi\rangle = |\psi\rangle\,, \forall S \in \mathcal{S}\}$$

is called a **stabilizer code** with stabilizer $\mathcal{S}$.

Clearly, in the qubit case, instead of requiring $iI \notin \mathcal{S}$, we ask for $-I \notin \mathcal{S}$, otherwise, if $-I \in \mathcal{S}$ the code subspace would be empty. This stabilizer definition simplifies a lot the study of quantum error-correcting codes and encompasses a wide class of QECCs. Not all stabilizers will give rise to good quantum error-correcting codes. First, they could have a very small code rate, which means that they encode a tiny amount of logical qubits on an exceedingly vast amount of physical qubits. Second, even if they could have a high code rate, they can be awful at correcting errors if the distance of the code is low. These are the things we must be careful of when working with stabilizer codes, and we must find a good stabilizer that finds a balance between code rate and distance.

To calculate code rates in stabilizer codes, we must find a minimal set of generators G of the stabilizer $\mathcal{S}$. If G has size r, then the code $T(\mathcal{S})$ encodes $k = n - r$ logical qudit in n. To calculate how good a code is, we must find what errors it can detect. The normalizer of $\mathcal{S}$

$$N(S) := \{N \in \mathcal{P}_n \mid \forall M \in S, [M,N] = 0\}$$

is the set of operators that cannot be detected by the stabilizer S. Since $S \subseteq N(S)$, the set of errors that cannot be detected with S is $N(S) \setminus S$.

Theorem 2.1.5. *Let S be the stabilizer of a stabilizer code. Let $\mathcal{E} \subseteq \mathcal{P}_n$ be a set of errors. Then, the errors $\mathcal{E}$ can be corrected by the code if*

$$E_i^\dagger E_j \notin N(\mathcal{S}) \setminus \mathcal{S}, \quad \forall E_i, E_j \in \mathcal{E}.$$

From this theorem, we can conclude that the distance d of $T(\mathcal{S})$ is the minimum weight of the operators in $N(\mathcal{S}) \setminus \mathcal{S}$. We denote a distance d stabilizer code which encodes k logical qubits in n physical qubits as an $[[n, k, d]]$ quantum code.

The Stabilizer formalism allows a brief description of specific codes. A relevant code that we mentioned above is the Shor code, a nine-qubit code, and the first proposed quantum error-correcting code that enables the correction of any single-qubit error. This code is a stabilizer code with the following generating set:

$$\begin{aligned}
&Z \otimes Z \otimes I \otimes I \otimes I \otimes I \otimes I \otimes I \otimes I \\
&Z \otimes I \otimes Z \otimes I \otimes I \otimes I \otimes I \otimes I \otimes I \\
&I \otimes I \otimes I \otimes Z \otimes Z \otimes I \otimes I \otimes I \otimes I \\
&I \otimes I \otimes I \otimes Z \otimes I \otimes Z \otimes I \otimes I \otimes I \\
&I \otimes I \otimes I \otimes I \otimes I \otimes I \otimes Z \otimes Z \otimes I \\
&I \otimes I \otimes I \otimes I \otimes I \otimes Z \otimes I \otimes Z \\
&X \otimes X \otimes X \otimes X \otimes X \otimes X \otimes I \otimes I \otimes I \\
&X \otimes X \otimes X \otimes I \otimes I \otimes I \otimes X \otimes X \otimes X.
\end{aligned}$$

From this set of generator we can calculate two things: the code rate and the distance. The code rate is $1/9$ since $k = 9 - 8 = 1$ and $n = 9$. The distance is the weight of the minimum weight error in $N(\mathcal{S}) \setminus \mathcal{S}$ which in this case is 3.

Another example of an important stabilizer code with fewer physical qubits and with the property of correcting any single-qubit error is the $[[7, 1, 3]]$ Steane code. Its stabilizer is generated by

$$\begin{aligned}
&Z \otimes Z \otimes Z \otimes Z \otimes I \otimes I \otimes I \\
&Z \otimes Z \otimes I \otimes I \otimes Z \otimes Z \otimes I \\
&Z \otimes I \otimes Z \otimes I \otimes Z \otimes I \otimes Z \\
&X \otimes X \otimes X \otimes X \otimes I \otimes I \otimes I \\
&X \otimes X \otimes I \otimes I \otimes X \otimes X \otimes I \\
&X \otimes I \otimes X \otimes I \otimes X \otimes I \otimes X,
\end{aligned}$$

and the logical operators are

$$\begin{aligned}
\bar{X} &= X \otimes X \otimes X \otimes X \otimes X \otimes X \otimes X \\
\bar{Z} &= Z \otimes Z \otimes Z \otimes Z \otimes Z \otimes Z \otimes Z.
\end{aligned}$$

It can correct arbitrary errors on single qubits. The first three elements of the stabilizer can detect any X error, and the last three elements of the stabilizer can detect any Z error. Since it can correct X and Z errors on single qubits, it can correct single Y errors. Ergo, every single-qubit error can be corrected because $\{I, X, Z, Y\}$ is a basis for the 2×2 Hermitian matrices. Additionally, it can correct errors of the

form $X_i \otimes Z_j$ for $i \neq j$. This is because X_i commutes with the last three elements of the stabilizer but anti-commute with the first three, and Z_j commutes with the first three and anti-commute with the last three.

To be sure that these errors can be corrected, they all must have different error syndromes. We can check that any single X_i error will correspond to a unique non-trivial syndrome on the first three stabilizers. Similarly, for single Z_i errors, they will correspond to unique non-trivial syndromes on the last three stabilizers. The Y_i errors are iX_iZ_i errors so that the syndrome will be the syndrome of X_i plus the syndrome of Z_i. Similarly, $X_i \otimes Z_j$ have unique syndromes, and they are the syndrome of X_i plus the syndrome of Z_j. Additionally, any linear combination of the mentioned errors will also be correctable with this 7-qubit code by properties of error-correcting codes (the state will collapse to one of those errors upon the measurement of the ancillas).

The syndrome measurement of this code can be performed as shown in Figure 2.1.

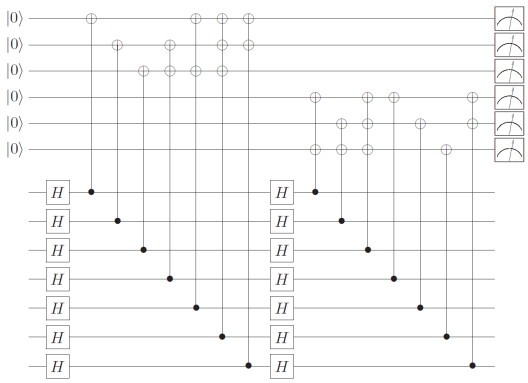


FIGURE 2.1: Error syndrome extraction of the Steane code. Taken from [18].

Assuming that we can perfectly encode a single qubit state, perform error correction and decode the state (which, in reality, is not the case), let us find the probability threshold of the depolarizing quantum channel to reduce the probability of error arbitrarily. This quantum channel is the superoperator that acts as follows

$$\rho \mapsto (1-p)\rho + \frac{p}{3}\left(X\rho X + Y\rho Y + Z\rho Z\right).$$

Encoding the initial state using the 7-qubit code, we obtain a state in 7 qubits σ which under the error model is modified as

$$\sigma = \rho_1 \otimes \cdots \otimes \rho_7 \to (1-p)^7\sigma + \binom{7}{1}(1-p)^6 p\mathcal{E}_1(\sigma) + \binom{7}{2}(1-p)^5 p^2\mathcal{E}_2(\sigma) + \cdots$$

where $\mathcal{E}_k(\sigma)$ denotes a state where k qubits suffered an error. In the simplified model that we are considering where we can perform a perfect syndrome extraction, error correction, and no need for magic state distillation, all single-qubit errors can be fixed and some two-qubit errors as mentioned above. This means that after a recovery $\mathcal{R}$, we will have the state

$$\mathcal{R}(\mathcal{E}(\sigma)) = \left((1-p)^7 + \binom{7}{1}(1-p)^6 p + \binom{7}{2}(1-p)^5\left(\frac{2p^2}{9}\right)\right)\sigma + \binom{7}{2}(1-p)^5\left(\frac{7p^2}{9}\right)\mathcal{E}_2(\sigma) + \cdots$$

Thus, if we want to decrease the error probability by concatenating this code in a tree-like structure, we must have that

$$f(p) := \left((1-p)^7 + \binom{7}{1}(1-p)^6 p + \binom{7}{2}(1-p)^5\left(\frac{2p^2}{9}\right)\right) > (1-p).$$

This happens when $p < 0.0764 \approx p_{th}$. Then, concatenating l layers of the Steane code will reduce the logical qubit error probability from p to $(1-f)^l(p)$. We can visualize this behavior in Figure 2.2.

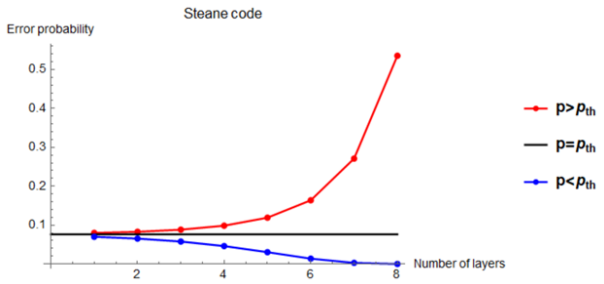


FIGURE 2.2: Probability of error after multiple encoding layers for $p < p_{th}$ and for $p > p_{th}$.

If we want to include gates and syndrome extraction errors, we would need to account for them by including their action as Kraus operators of the quantum channel. A similar analysis to find the probability threshold on *fault-tolerant computing* is done in Nielsen's book [18]. In this case, we would need to consider all possible errors on the ancillas, errors on the single-qubit and two-qubit gates, and errors on measurements and prevent them from spreading between encoded blocks.

2.1.3 Fault Tolerance

Quantum error correction works on an idealized scenario where gates between qubits are perfect. In real life, this does not happen. A quantum process (such as quantum computing) can have a failure in many places, from preparing a state and applying gates to measuring Pauli operators. Despite all of these problems afflicting quantum processes, there is a way of getting around them. Fault tolerance deals with these errors and reduces their probability of occurrence. The idea is to find a procedure that, given a circuit C, we obtain a new circuit $\tilde{C}$, a fault-tolerant encoding of C. We must map each part of a circuit, from state preparation to error correction to its fault-tolerant encoding. Usually, the fault-tolerant encoded components are referred to as *fault-tolerant gadgets* [12]. To understand the ideas behind fault tolerance, let us show the basic model of fault tolerance. This model is very inefficient but shows a solid theoretical result for quantum computing, and luckily this result can also be adjusted to a more realistic scenario.

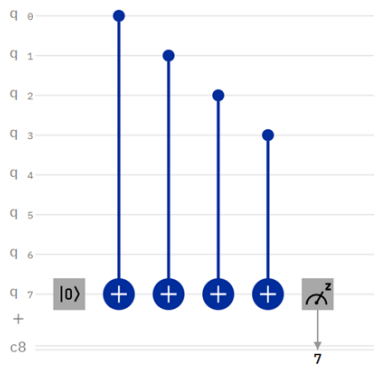


FIGURE 2.3: Circuit for the syndrome measurement of $Z \otimes Z \otimes Z \otimes Z \otimes I \otimes I \otimes I$.

First, consider measuring the stabilizer $Z \otimes Z \otimes Z \otimes Z \otimes I \otimes I \otimes I$ of the seven qubit QECC. The circuit that is usually used is the one shown in Figure 2.3. A problem with this circuit is that the ancilla state is coupled with multiple components of the encoded qubit. Therefore, if it has an error, such as a rotation, it will propagate to the four qubits on which the CNOT gates act. This detail shows how one error could build up into many more errors, somehow creating an avalanche of errors and completely ruining some computation. A way of dealing with it is using a cat state $|0\rangle^{\otimes 4} + |1\rangle^{\otimes 4}$ as an ancilla. Instead of coupling all the CNOTs to just one qubit, we now couple the individually to one of the four qubits of the cat state. For a small enough error probability p, this procedure lowers the chances that one error in the ancilla propagates and becomes two or more errors in the state that we care about. Now, we should be certain that the fault-tolerant ancilla (cat state) has no errors.

A way of confidently preparing a cat state, in our case $|0\rangle^{\otimes 4} + |1\rangle^{\otimes 4}$, is preparing it as usual and then doing a parity check as shown in Figure 2.4.

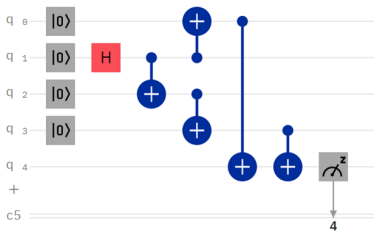


FIGURE 2.4: Circuit that prepares a cat state.

We use a normal ancilla qubit to measure if the cat state has an error or not. If it has an error, one can fix it or discard the state and start over again to prevent having errors. Although this circuit is not fault-tolerant, it prevents the errors from spreading. Therefore, this shows a simple, although inefficient, way of performing fault-tolerant syndrome measurement.

Next, we must show that we can also do error correction, measurements, state preparation, and gates fault tolerantly. Shor [36] cleverly addressed these problems. He proposed to do fault-tolerant error correction by repeating the error syndrome measurement procedure enough times to be confident that the answer is most likely correct:

1. Measure the elements of the stabilizer using a cat state.

2. Repeat the syndrome measurement $(t+1)^2$ times.

3. Correct the error that is associated with the the last and most repeated error syndrome.

Similarly, to perform a fault-tolerant measurement of a Pauli operator P, the procedure is to

1. Do Shor fault-tolerant error correction.

2. Use a cat state to measure the Pauli operator P.

3. Repeat $2t+1$ times the previous two steps, and as the fault-tolerant measurement outcome, pick the most repeated measurement outcome.

Furthermore, performing a fault-tolerant state preparation of the state $|0\rangle$ is relatively easy since we know how to do FT error correction. To do so, we need to do error correction of the stabilizer $\langle \mathcal{S}, \bar{Z} \rangle$ where $\mathcal{S}$ is the stabilizer of a chosen error-correcting code and $\bar{Z}$ is the logical Z operator of the code. This happens because the state $|0\rangle$ is the only logical state of a code that is both stabilized by $\bar{Z}$ and by $\mathcal{S}$. The mentioned procedures deal with performing error correction, measurement, and state preparation in a fault-tolerant manner. Still, we have not yet mentioned the most crucial step in quantum computation: applying gates.

To have the capabilities of an ideal quantum computer, we need access to a universal set of gates that arbitrarily approximate any quantum operation. A common way of achieving universality is choosing a set of gates that generate the *Clifford group* and add any other gate that doesn't belong in that group [37].

Definition 2.1.7. The **Clifford group** is the normalizer of the Pauli group

$$\mathbf{C}_n = \{T \in U(2^n) \mid T\mathcal{P}_n T^\dagger = \mathcal{P}_n\}.$$

A set of gates that generate $\mathbf{C}_n$ is

$$\mathbf{C}_n = \text{gen}\left\{H, \text{CNOT}, \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/2} \end{pmatrix}\right\}.$$

If we only had access to implementing the gates $\left\{H, \text{CNOT}, R_{\pi/4} = \text{diag}(1, e^{i\pi/2})\right\}$, we will not achieve any quantum advantage. A fundamental theorem that quantifies how powerful a quantum computer would by having access to a chosen set of gates is the **Gottesman-Knill theorem** [38]. It states that any circuit consisting of gates from the Clifford group can be efficiently simulated (in polynomial time) on a probabilistic classical computer [39]. So, to achieve a universal set of gates, we must also implement a gate which does not belong to $\mathbf{C}_n$, like $R_{\pi/8} = \text{diag}(1, e^{i\pi/4})$.

But how can we implement these gates without propagating errors all around the qubits? A solution to this is using *transversal gates* [12]. A **transversal gate** between two code blocks is a gate that only interacts the i-th qubits of one block with the i-th qubits of the other block. For example, for the seven-qubit code, the implementation of the CNOT gate can be performed transversally between two code blocks (each of 7 qubits) as shown in Figure 2.5.

Similarly, the transversal implementation of the logical Hadamard gate in the seven-qubit code is

$$\bar{H} = H^{\otimes 7},$$

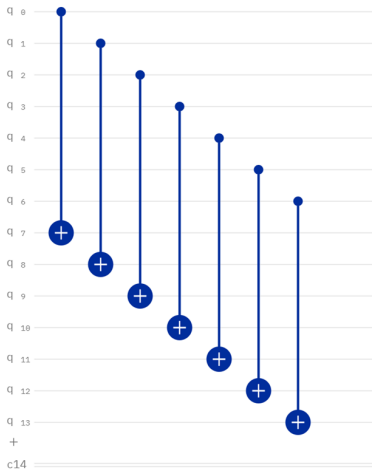


FIGURE 2.5: A logical CNOT transversal gate between two seven-qubit blocks.

and of the logical rotation by $\pi/4$ is

$$\bar{R}_{\pi/4} = R_{-\pi/4}^{\otimes 7} = \left(\text{diag}(1, e^{-i\pi/2})\right)^{\otimes 7}.$$

These are enough gates to generate all the Clifford gates, and in fact, any CSS code can implement Clifford gates as transversal gates. Finally, to have a universal set of gates that achieve true quantum advantage, we must perform in a fault-tolerant way a non-Clifford gate like $R_{\pi/8}$. Unfortunately, these gates cannot be implemented transversally. Chen et al. showed that it is impossible to have a universal transversal set of gates for stabilizer codes [40]. Despite this result, a way of getting around it is employing *magic states*.

Consider the following setup. You want to implement a gate $U \in SU(2^d) \setminus \mathbf{C}_n$ such that $\forall P \in \mathcal{P}_n, UPU^\dagger \in \mathbf{C}_n$, for example $R_{\pi/8}$, preventing the uncontrolled propagation of errors. To do so, we use a procedure called *magic state injection*, inspired by the standard teleportation procedure of qubits [41]. The standard teleportation protocol begins with two parties Alice and Bob, both sharing an entangled pair of qubits. Alice sends Bob a state $|\psi\rangle$ by doing a Bell measurement on both $|\psi\rangle$ and the entangled qubit she has and then sends the classical output to Bob. With this information, Bob can recover $|\psi\rangle$. If Bob wants to have the state $U|\psi\rangle$, he will apply the gate U on his qubit after the teleportation protocol.

The idea of magic state injection is to propagate the U gate through the conditional Pauli operator Bob uses after receiving Alice's classical message. This Pauli P will be updated to $U^\dagger PU$ which is, by construction, a Clifford group operation. Note that the gate $U^\dagger PU$ can be implemented fault tolerantly, but still, the gate U cannot. Joining both the entangled pair and the U gate, we obtain a new state $(I \otimes U)|Bell\rangle$, also known as a magic state. Generally speaking, **magic states** are quantum states prepared beforehand in a quantum computation to be later used for implementing a non-Clifford gate. The trouble of implementing a non-Clifford gate fault-tolerantly is translated into preparing with high fidelity a particular magic state. An easy but inefficient solution to preparing these states is using the Shor state preparation protocol discussed above. The state preparation would be done using Shor error correction, where the stabilizer is the transformed Bell state stabilizer. In the case of a two-qubit Bell state whose stabilizer is $\{X \otimes X, Z \otimes Z\}$, when we propagate the U gate thought

the second qubit, we get the updated stabilizer $\{X \otimes UXU^\dagger, Z \otimes UZU^\dagger\}$. Nowadays, a more efficient procedure for preparing these states is called magic state distillation. As the name suggests, the idea of it is obtaining a magic state with high fidelity out of multiple noisy magic states.

Finally, having achieved a fault-tolerant map that encodes every part of a circuit into a fault-tolerant gadget, we can jump to the main result of fault tolerance, the threshold theorem.

Theorem 2.1.6 (Threshold theorem [18])**.** *A quantum circuit containing $p(n)$ gates may be simulated with probability of error at most ε using*

$$O(\text{poly}(\log p(n)/\epsilon)p(n))$$

gates on a hardware given that the error rate p is below some threshold value p_{th}.

This theorem in essence, as Scott Aranson describes it, states that you can correct errors faster than the speed at which they occur, in principle, achieving a computation with arbitrary precision.

2.2 AME States and Quantum Error Correction

Up to this point, we have covered the basis for quantum error correction. We can now shift our focus to one of our main discussion points of this thesis, the building blocks of holographic codes: AME states. The primary references that we will follow in this subsection are Helwig's articles [10, 11, 16], which I recommend if the reader wants to have more details on this topic.

2.2.1 AME States

Absolutely maximally entangled states, or *AME states*, are highly entangled states that have multiple applications in quantum error correction [13], quantum communication [11], and quantum cryptography [10]. Before defining AME states, let us explain what a *maximally entangled bi-partition* is [18].

Definition 2.2.1. Given a state of n qudits (of dimension d), a non-trivial bi-partition of the system $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$, where A and B are non empty sets of qudits and $m = |B| \leq |A| = n - m$, is said to be **maximally entangled** if it satisfies that

$$S(\rho_B) = m \log_2 d,$$

where ρ_B is the reduced state to the subsystem B. This condition of maximal entanglement can be checked with other equivalent criteria:

Lemma 2.2.1. *For a pure state $|\psi\rangle$ of n qudits (dimension d) and a bipartition $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$ of the n qudits into sets A and B, such that $m = |B| \leq |A| = n - m$. The following conditions are equivalent:*

- $S(\rho_B) = m \log_2 d$.
- $\rho_B = \text{Tr}_A |\psi\rangle\langle\psi| = \frac{1}{d^m} I_B$.
- $|\psi\rangle = \frac{1}{\sqrt{d^m}} \sum_{k \in \mathbb{Z}_d^m} |k_1\rangle_{B_1} \cdots |k_m\rangle_{B_m} |\phi(k)\rangle_A$ *where* $\langle \phi(k) \mid \phi(k')\rangle = \delta_{kk'}$.

Basically, maximally entangled states are a generalization of Bell states. With this characterization of maximally entangled states in mind, we can now define AME states:

Definition 2.2.2. An **AME(n, d) state** is a pure state of n qudits such that all non-trivial bi-partitions of the n qudits are maximally entangled.

This definition might seem quite restrictive at first glance, and indeed it is. For arbitrary values of n and d, AME states might not exist. For example, there are no AME states for $(n = 4, d = 2)$, $(n = 7, d = 2)$, and for many other values of n and d [42, 43, 44] . Still, there is a way of constructing some of them for arbitrary n as long as we do not have restrictions over the dimension d. Helwig showed [11] that we can use MDS classical codes to construct AME states in the following way:

- An $AME(2m, d)$ state can be constructed from a MDS code $\mathcal{C}$ of length $2m$ and minimal distance $m + 1$ over an alphabet of d letters as

$$|\psi\rangle = \frac{1}{\sqrt{d^m}} \sum_{c\in\mathcal{C}} |c_1\rangle_1 \cdots |c_m\rangle_m |c_{m+1}\rangle_{m+1} \cdots |c_{2m}\rangle_{2m} .$$

- An $AME(2m + 1, d)$ state can be constructed from a MDS code $\mathcal{C}$ of length $2m + 1$ and minimal distance $m + 2$ over an alphabet of d letters as

$$|\psi\rangle = \frac{1}{\sqrt{d^m}} \sum_{c\in\mathcal{C}} |c_1\rangle_1 \cdots |c_m\rangle_m |c_{m+1}\rangle_{m+1} \cdots |c_{2m+1}\rangle_{2m+1} .$$

As we mentioned in the previous Chapter, after choosing a basis, we can view a quantum state, particularly, a qudit state, as a tensor T. AME states correspond to a family of tensors called *perfect tensors*[1].

Definition 2.2.3. A **perfect tensor** T is a tensor that for any bipartition A and B ($|A| \leq B$) of its indices is proportional to an *isometry* from A to B. This means that the map $T_A : \mathcal{H}_A \to \mathcal{H}_B$ defined by

$$T_A(|a\rangle) := \sum_b T_{ab} |b\rangle ,$$

preserves the inner product. Here, the sets $\{|a\rangle\}_{a\in A}$ and $\{|b\rangle\}_{b\in B}$ are orthonormal basis for $\mathcal{H}_A$ and $\mathcal{H}_B$, respectively.

Note that T_A preserving the inner product is equivalent to saying that the tensor T satisfies that $\sum_b T^\dagger_{ba} T_{\tilde{a}b} = \delta_{a\tilde{a}}$. From Lemma 2.2.1, we get the correspondence between AME states and perfect tensors. Every bi-partition of AME states is maximally entangled, meaning that the state can be written as

$$|\psi\rangle = \frac{1}{\sqrt{d^m}} \sum_{k\in\mathbb{Z}_d^m} |k_1\rangle_{A_1} \cdots |k_m\rangle_{A_m} |\phi(k)\rangle_B,$$

where $\langle\phi(k) \mid \phi(k')\rangle = \delta_{kk'}$, giving rise to an isometric map $T_A : \mathcal{H}_A \to \mathcal{H}_B$ defined by $T_A(|a\rangle) := \langle a|\psi\rangle = \frac{1}{\sqrt{d^m}} \sum_{k\in\mathbb{Z}_d^m} \langle a|k\rangle_A |\phi(k)\rangle_B$.

Some of the simplest examples of AME states are the GHZ state and the Bell states. For the next chapter on holographic codes, we will use the $AME(5, 2)$ and $AME(6, 2)$ states [45]

[1] Note that perfect tensors and perfect codes are two different notions.

$$\begin{aligned}|AME(5,2)\rangle =& \frac{1}{4}(|00000\rangle + |10010\rangle + |10100\rangle + |01010\rangle - |11011\rangle - |00110\rangle \\ & - |11000\rangle - |11101\rangle - |00011\rangle - |11110\rangle - |01111\rangle - |10001\rangle \\ & - |01100\rangle - |10111\rangle + |00101\rangle),\end{aligned}$$

$$|AME(6,2)\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |AME(5,2)\rangle + |1\rangle \otimes (X^{\otimes 5}) |AME(5,2)\rangle ,$$

for constructing a larger family of states. Furthermore, to obtain the stabilizer of quantum error-correcting codes created from AME states we need to introduce a larger family of quantum states called *graph states*.

2.2.2 Graph States

As the name suggests, graph states are quantum states constructed from a graph. An undirected graph G is a set of n vertices and weighted edges between vertices. We can obtain this graph's adjacency matrix $A \in \mathbb{Z}_d^{n\times n}$, a matrix with entry $A_{ij} \in \mathbb{Z}_d \setminus \{0\}$ if there is an edge with weight A_{ij} connecting the points i and j and 0 otherwise.

Definition 2.2.4. Given a graph G with an adjacency matrix A, the associated qudit **graph state** is defined as

$$|G\rangle = \prod_{i>j} CZ_{ij}^{A_{ij}} \left(F^\dagger |0\rangle\right)^{\otimes n},$$

where $F^\dagger |p\rangle = \frac{1}{\sqrt{d}} \sum_{k=0}^{d-1} \omega^{-kp} |k\rangle$ is the Hermitian adjoint of the quantum Fourier transform.

Graph states can be thought of as quantum codes encoding zero logical qubits (they are stabilizer states). From this perspective, they have a simple stabilizer given by the following set of generators:

$$\left\{ X_i \prod_{j\in\{1,\ldots,n\}} Z_j^{A_{ij}} \right\}_{i\in\{1,\ldots,n\}}.$$

An example is the qubit graph state obtained from the graph G whose shape is a pentagon's perimeter (Figure 2.6).

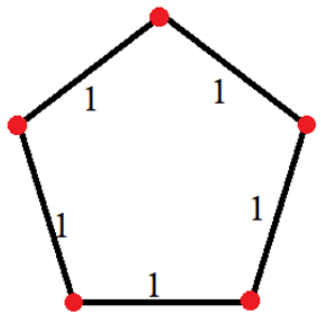


FIGURE 2.6: A graph with 5 vertices used to create a graph state.

The adjacency matrix of G is

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix},$$

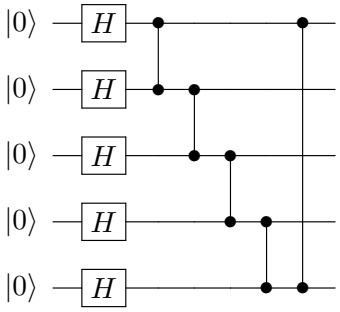
and the associated graph state is

$$|G\rangle = CZ_{12}CZ_{14}CZ_{23}CZ_{34}CZ_{45}(|+\rangle \otimes |+\rangle \otimes |+\rangle \otimes |+\rangle \otimes |+\rangle).$$

At first glance, these states might seem arbitrary, but they cover a wide spectrum of entangled states. Bahramgiri and Beig showed [46] that every stabilizer state is equivalent to some graph state modulo Clifford group gates. Therefore, the study of an entangled stabilizer state is simplified to studying properties of the adjacency matrix associated to the equivalent graph state. Some entanglement properties from a graph state can be derived right off the graph itself, like knowing if a state is AME or not, but still, it is not always straightforward how to do so. Helwig proved [16] a theorem that shows a way of determining if a graph state is AME or not by using its adjacency matrix:

Theorem 2.2.2. *A graph state $|G\rangle$ with adjacency matrix A is an AME state if and only if for all $K \subseteq \{1, \ldots, n\}$, such that $|K| = \lfloor \frac{n}{2} \rfloor$, the vectors $\{A_{k_i} \setminus K\}_{k_i \in K}$ are linearly independent in $\mathbb{Z}_p^{n-m}$, where $A_{k_i} \setminus K$ is the k_i row of A with the entries $\{A_{k_i k_j}\}_{k_j \in K}$ removed.*

This relationship between AME states and graph states gives us a direct protocol, straightforward from the definition of graph states, for constructing AME states on quantum circuits. For example, the graph from the perimeter of a pentagon (Figure 2.6) is an $AME(5, 2)$ state. Therefore, to build this AME state, we use the definition of the graph state and obtain the following circuit: [45]:



2.2.3 Quantum Codes from AME States

To conclude this chapter, we can now study the applications that AME states have, particularly in quantum error correction. We can construct quantum codes from AME states by tracing out qubits. A single AME state is equivalent to a $[[N, 0, \lfloor N/2 \rfloor + 1]]$ quantum code, and from it we can construct a set of $\lfloor N/2 \rfloor$ additional quantum codes:

$$\{[[N - i, i, \lfloor N/2 \rfloor + 1 - i]]\}_{i \in \{1, \ldots, \lfloor N/2 \rfloor\}}.$$

Let us consider one of the AME states we mentioned before,

$$|AME(6,2)\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |AME(5,2)\rangle + |1\rangle \otimes (X^{\otimes 5})\,|AME(5,2)\rangle\,.$$

This is a $[[6,0,4]]$ quantum code, and by tracing out the first qubit, we obtain the 5-qubit $[[5,1,3]]$ perfect code, whose logical qubits are:

$$\begin{aligned}|\bar{0}\rangle &= |AME(5,2)\rangle\\ |\bar{1}\rangle &= (X^{\otimes 5})\,|AME(5,2)\rangle\,.\end{aligned}$$

The procedure is the same to obtain the other two quantum codes; one needs to trace out more qubits to increase the dimension of the logical space. To have a description of these codes in terms of a stabilizer, we use the result by Bahramgiri, and Beig [46] mentioned in the previous subsection. Each AME state is equivalent (modulo Clifford group operators) to a graph state that has an explicit generating set of the stabilizer. Using the m generators of the graph state's stabilizer, we can obtain the stabilizer of the new quantum codes. As shown by Mazurek et al. [13], this is achieved by first doing elementary operations on the generator matrix to transform it in a way such that the last $n-m$ columns of every row are weight one (X or Z) or weight zero operators, and then removing the first $2k$ generator rows as well as the last k columns where k is the number of logical qubits of the quantum code. For example, the stabilizer of the $AME(6,2)$ is [13]:

$$\begin{bmatrix} X & Z & Z & X & 1 & 1\\ 1 & X & Z & Z & X & 1\\ X & 1 & X & Z & Z & 1\\ Z & X & 1 & X & Z & 1\\ X & X & X & X & X & X\\ Z & Z & Z & Z & Z & Z \end{bmatrix},$$

and it can be transformed using elementary operations to

$$\begin{bmatrix} -Y & Z & Y & 1 & 1 & Z\\ -Z & X & Z & 1 & 1 & X\\ Y & Y & Z & 1 & Z & 1\\ Z & Z & X & 1 & X & 1\\ -Z & Y & Y & Z & 1 & 1\\ X & Z & Z & X & 1 & 1 \end{bmatrix}.$$

Then, to obtain the $[[5,1,3]]$ perfect qubit code, we remove the first two rows and the last column to obtain the stabilizer generator of the code:

$$S = \langle Y\otimes Y\otimes Z\otimes I\otimes Z, Z\otimes Z\otimes X\otimes 1\otimes X, Z\otimes Y\otimes Y\otimes Z\otimes 1, X\otimes Z\otimes Z\otimes X\otimes 1\rangle.$$

Doing this procedure is helpful to study these codes since the stabilizer formalism significantly simplifies working with quantum codes. We can easily calculate error syndromes and code properties such as the distance and the code rates.

Finally, other interesting applications of AME states, beside producing quantum codes, in quantum cryptography and quantum communications are:

- **Quantum Secret Sharing Schemes [10]:**

  A Secret Sharing Scheme is a procedure on which a party, usually called the dealer, gives pieces of information to other parties. This information can be used to recover a secret that the dealer encoded. Specifically, an (n, k)-threshold secret sharing scheme is an instance of a sharing scheme whereby having the information of any k parties of the total n parties, the secret can be fully recovered and could not be retrieved by having less than the knowledge of k parties. An (n, k)-quantum threshold secret sharing scheme (QSS) is a scheme on which, by using a quantum state distributed among the n parties, a threshold secret sharing scheme can be accomplished. Helwig proved [10] a one to one correspondence between $AME(2m, d)$ states and $((m, 2m-1))$ QSS, where the QSS can be achieved by distributing each qudit on an $AME(n, d)$ state to each different party.

- **Quantum Teleportation [11]:**

  This is the procedure on which a party (Alice) instantly sends a quantum state to another party (Bob) and later sends a classical set of bits for the second party to recover the initial quantum state. This is possible if Alice and Bob previously share a Bell state. AME states are maximally entangled states for any bi-partition of its qudits. This implies that if two sets of qubits $\{A, B\}$ were distributed among two parties (Alice and Bob), they could perform local operations to convert the maximal entanglement in Bell states. Both parties would share n Bell states, where $n = \min\{|A|, |B|\}$ and they can be used to teleport a 2^n dimensional quantum state from one party to the other. Helwig proved [11] in more complex scenarios that this teleportation could be possible even if a subset (A or B) has its qubits isolated, meaning it is not able to perform 2-qubit gates within its qubits.

Chapter 3

Holography and QEC

3.1 Holography

The purpose of this third chapter is to study holographic codes. We first introduce the AdS/CFT correspondence as well as some relevant results from this field. Then, we motivate the creation of holographic codes and study their relationship with the AdS/CFT correspondence. Finally, we calculate some code rates for the obtained quantum codes and discuss if they are appropriate for quantum processing tasks. We mainly follow the first articles on holography and error correction by Pastawski et al.[8, 9] and a recent review on this topic by Jahn et al [28].

3.1.1 The AdS/CFT Correspondence

Our universe is filled with extraordinary objects, and black holes are on top of the list. These are thermodynamic objects with a defined entropy and temperature. But contrary to physical intuition, the black hole entropy depends on the surface area and not the volume. In some way, this fact means that the microstate information of the black hole is somehow encoded on its boundary. Bekenstein showed [47] that our universe has an upper limit on the amount of information that can be stored in a region of space. This limit is called the **Bekestein bound**, and it limits the entropy of a region of space before it becomes a black hole

$$S_{\text{BH}} = \frac{A_{\text{horizon}} c^3}{4\hbar G} = \frac{A_{\text{horizon}}}{4\ell_p^2}, \quad S \leq \frac{2\pi k R E}{\hbar c}.$$

The equation on the left is the Bekenstein-Hawking entropy, and the one on the right is the Bekenstein bound. Here ℓ_p is the plank length, G is the gravitational constant, c the speed of light, E is the mass-energy enclosed by a sphere of radius R, and k is the Boltzmann constant. This idea of reducing gravity dynamics on a $3+1$ dimensional space into a $2+1$ dimensional space is called the **holographic principle** [5], and every quantum gravity theory must satisfy it.

Despite this principle's vagueness, in 1997, Juan Maldacena showed [6] that this could be possible in the String theory setup. He conjectured that there is a correspondence between observables in an *Anti-de Sitter space*, a type of space-time with gravity, with operators in a *conformal field theory*, a quantum field theory without gravity. Nowadays, this correspondence is called the **AdS/CFT correspondence**. To understand this idea, let us break down each component:

Definition 3.1.1. An **Anti-de Sitter space** (AdS) is a maximally symmetric Lorentzian manifold with negative scalar curvature, and it is a solution to Einstein's field equations.

A *Lorentzian manifold* is a differentiable n-dimensional manifold that has a non-degenerate symmetric metric tensor (a way of measuring distance) with a metric signature of $(n-1,1)$, these are the type of manifolds that arise from solving Einstein's field equations. A manifold is called *maximally symmetric* if it allows $n(n+1)/2$ linearly independent Killing fields. There are multiple maximally symmetric Lorentzian manifolds that are solution to Einstein's field equations and have different curvatures. If it has positive curvature, the space is called a *de Sitter space*. If it has negative curvature, we have an AdS space. An example of an AdS space that we will stick to throughout this chapter is AdS_3, a $2+1$ space-time traced by a *Poincaré disk* as shown in Figure 3.1.

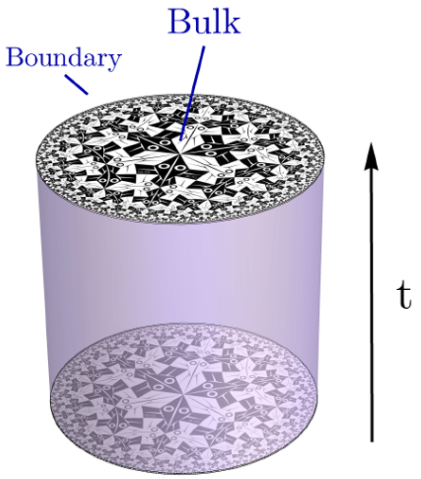


FIGURE 3.1: A representation of AdS_3 space, which is a Poincaré disk moving through time. The bulk refers to the AdS gravity theory, and the boundary refers to the CFT. Taken from [28].

Definition 3.1.2. A **conformal field theory** (CFT) is a quantum field theory invariant under conformal transformations.

A *conformal transformation* is an element of the *conformal group*, this is, the group of all the possible transformations from the space to itself that preserve the angles between vectors. This group contains familiar transformations, such as the maps from the Poincaré group; these are the mixture of Lorentz transformations and translations of space. Still, it contains a larger set of maps, like scale transformations $x^\mu \mapsto \lambda x^\mu$, and more. In general, conformal maps are coordinate transformations that locally rescale the metric of the space. A quantum field theory is said to be invariant under conformal transformations if all the *n-point functions* stay the same, before and after a conformal transformation. This means that for a set of fields $\{\phi_1, \ldots, \phi_n\}$ of the CFT and an element f of the conformal group with a group representation of π_f, we require that

$$\langle \phi_1(x_1) \cdots \phi_n(x_n) \rangle = \langle \pi_f(\phi_1)(x_1) \cdots \pi_f(\phi_n)(x_n) \rangle .$$

Finally, the AdS/CFT correspondence [6] is a relationship between an AdS space with a CFT living on its boundary. The former space is referred to as the *bulk*, and the latter as the *boundary*. Using the example of AdS_3, this correspondence relates gravity fields of the $2+1$ AdS space with operators of the $1+1$ CFT that is on its boundary. We can think of it as a dictionary between gravity dynamics in the AdS

space and the dynamics of the boundary CFT. Statistically speaking, the fields ϕ in the AdS spaces have operators $\mathcal{O}$ in the CFT whose partition functions, coincide

$$Z_{\text{bulk}}\left[\phi\right] = Z_{\text{boundary}}\left[\mathcal{O}\right].$$

In this case, the partition function of the boundary is given by

$$Z_{\text{boundary}}\left[\mathcal{O}\right] = \left\langle \exp\left(-\sum_i \int d^d x \phi_0^i(x)\mathcal{O}^i(x)\right)\right\rangle,$$

where ϕ_0 is the boundary condition for the fields $\phi^i(\partial M) = \phi_0^i(x)$, and the partition function of the bulk is given by

$$Z_{\text{bulk}}\left[\phi\right] = \int DgD\phi e^{-S[\phi]},$$

Where g is the metric of the space and $S[\phi]$ is the action on the AdS space. Overall, this correspondence is still not completely understood and, from a rigorous perspective, should be viewed with skepticism. The proposal of holographic codes aims to gain a different point of view for the AdS/CFT correspondence in order to explain some of the main aspects of it, as we will see later in the chapter.

3.1.2 RT Formula and AdS-Rindler Reconstruction

We will address two relevant properties of the AdS/CFT correspondence, the *Ryu and Takayanagi formula* [48] and the *Causal wedge (AdS-Rindler) reconstruction* [49]. These are inherent properties of this correspondence, and every model that tries to explain AdS/CFT should reproduce them. Ryu and Takayanagi showed that one could calculate the entanglement entropy of a conformal field theory from a holographic geometric description in the AdS/CFT duality[48, 17]. They argued that the entanglement entropy of a subregion of the CFT would be proportional to the minimal surface area in the AdS, whose boundary equals that of the subregion. This is a generalization of the Bekenstein-Hawking entropy formula, and surprisingly relates quantitatively a geometrical notion, the area of the minimal surface, with an information theoretic variable, the entropy. For a subregion A with minimal holographic surface γ_A, the **Ryu-Takayanagi formula** is

$$S_A = \frac{\text{Area}(\gamma_A)}{4G}, \tag{3.1}$$

where G is Newton's gravitational constant in the gravity theory. Figure 3.2 shows an illustration of these surfaces in the case of AdS_3.

On the other hand, the **AdS-Rindler reconstruction** states that for every point $x \in \mathcal{C}[A]$ inside the causal wedge of a subregion A of the CFT, where $C[A]$ is defined as all the bulk points which can both send or receive light signals from A, any bulk local operator $\phi(x)$ can be represented as a non-local CFT operator on A. This representation is indeed a precise map between both the operator $\phi(x)$ in the bulk and an operator $\mathcal{O}[\phi(x)]$ in the boundary. In Figure 3.3, the causal wedge encloses regions in the future and past of the considered timeslice of A. The argument for reconstructing the operator on only A lies in the fact that one can use the equations of motion of the boundary to find an explicit operator that encloses all the information required to reconstruct $\phi(x)$. Almheiri et al. [4] show that this reconstruction is non-unique, and there are many inequivalent operators on different boundary regions that

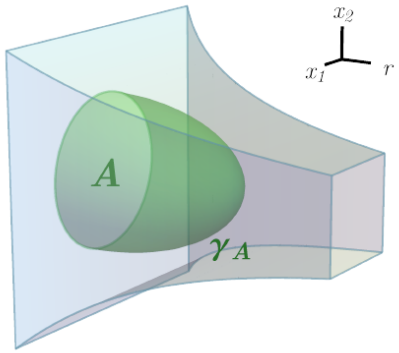


FIGURE 3.2: A sub-region A of the CFT with its corresponding minimal surface γ_A inside the bulk. Taken from [28]

allow obtaining $\phi(x)$. The idea is the following: first, we must note that since x can be in multiple causal wedges of different subregions, $\phi(x)$ can have numerous representations on the CFT. The reconstruction allows that for every pair of points x and y in the bulk and boundary, respectively, there exists an operator $\mathcal{O}[\phi(x)]$ which encodes the information of $\phi(x)$ and exist a small neighborhood U of y for which $\mathcal{O}[\phi(x)]$ has no support on. This implies that $[\mathcal{O}[\phi(x)], \hat{O}] = 0$ for all the operators $\hat{O}$ whose support is contained in U. Finally, we need to use the fact that the algebra of operators in the boundary is irreducible. Since y is arbitrary, if $\mathcal{O}[\phi(x)]$ was unique, by using *Schur's lemma* we would get that $\mathcal{O}[\phi(x)]$ must be the identity, which is a contradiction (otherwise, we would not be representing any observable in the boundary). Therefore, in AdS/CFT, a bulk local observable can be represented by many different operators in the CFT.

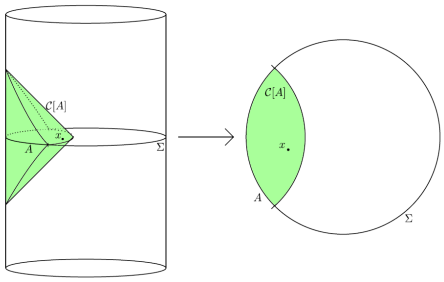


FIGURE 3.3: The green wedge is the causal wedge $\mathcal{C}[A]$ of the boundary sub-region A. Taken from [8].

3.2 Holographic Quantum Error-Correcting Codes

For the moment, holography has been described in a continuous setting. The motivation behind *holographic codes* is to obtain a discrete description of such relations prescribed by the holographic principle, particularly in the AdS/CFT context. Previous to these codes, discrete descriptions of holography were being studied. Swingel pointed out [50, 51] that the structure that arises from entanglement renormalization, such as the MERA tensor networks, could be considered a skeleton of a holographic duality. The MERA interpretation of holography provided similarities between entanglement entropy of tensor networks and the RT formula, but it did not reproduce

many other holographic properties. Recently, Almheiri et al. pointed out [4] that the redundancy on the causal wedge reconstruction, mentioned above, should be interpreted as a quantum error-correcting code. The reconstruction of an operator on a boundary region A should be thought of as the error correction of the erasure of the complementary boundary region A^c. It is because on the AdS-Rindler reconstruction of a bulk operator $\phi(x)$ over a boundary region A, one does not have access to the complementary region A^c, similar to the erasure of a set of qubits in quantum error correction. This detail inspired creating a large family of quantum codes with a MERA-like structure, referred to as holographic codes.

3.2.1 Holographic States and Codes

There have been many proposals for holographic codes [52, 8], but they have all converged to the family of codes proposed by Pastwski et al. [8]. These are codes that mimic hyperbolic space by placing perfect tensors (discussed in Chapter 2) on a hyperbolic tiling. These tensors are chosen so that it is easy to build isometric maps from different regions in the bulk to the boundary. Let us define what holographic states and codes are:

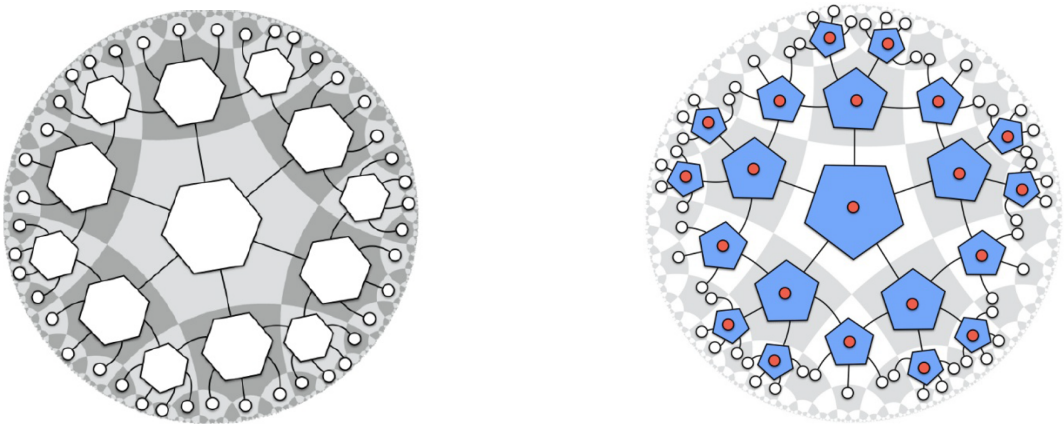
Definition 3.2.1. A **holographic state** is the state interpretation of a tensor network composed of perfect tensors covering a geometric manifold with a boundary where all interior legs are contracted. All uncontracted legs (physical degrees of freedom) are at the boundary of such manifold.

To explicitly build holographic states, we can use AME states, since they are in correspondence with perfect tensors as mentioned in the previous chapter. An example of a holographic state composed of thirteen $AME(6,2)$ states is shown in Figure 3.4a. Holographic states can be thought of as $[[n,0,d]]$ quantum codes encoding zero qubits, where the physical space is the holographic state itself. To define a code that has a non-trivial logical space, we need to have uncontracted legs in the bulk:

Definition 3.2.2. A **holographic code** is a tensor network composed of perfect tensors covering some geometric manifold with boundary if it is an isometry from uncontracted bulk legs to uncontracted boundary legs.

A relevant holographic code is the one proposed by Pastawski et al. [8] known as the *HaPPY code* (Figure 3.4b). It is composed of $AME(6,2)$ like the state in Figure 3.4a, but this time, one of the tensor legs of every $AME(6,2)$ state is left uncontracted for it to be interpreted as part of the bulk space.

We can construct more holographic codes similar to the HaPPY code by placing one tensor on the tiles of a uniform hyperbolic tessellation of the Poincaré disk (Figure 3.1). Uniform tessellations refer to tessellations of the space using equiangular and equilateral polygons with the property that every vertex of the tessellation has the same number and type of polygons around it. These uniform tessellations can be uniquely labeled with the so-called **Schläfli symbol** $\{p,q\}$. It represents an p-gon tiling where q p-gons meet at each vertex. A uniform hyperbolic tiling requires that $a(p,q) = pq - 2(p+q)$ is positive. When $a(p,q)$ is 0 or negative, $\{p,q\}$ corresponds to tiles in a flat space or polytopes, respectively. For the HaPPY code, $p = 4$ and $q = 5$, satisfying that $a(4,5) = 2$ is positive. The definitions of holographic states and holographic codes hold for higher dimensions of bulk and boundary. For higher-dimensional AdS spaces, holographic codes can be constructed from other hyperbolic tessellations. For example, for three-dimensional hyperbolic space, the *uniform honeycombs* are tessellations of $\mathbb{H}^3$ by uniform polyhedral cells. We would place perfect



(A) (B)

FIGURE 3.4: (A) An example of a holographic state constructed by concatenating six-index perfect tensors. (B) A holographic code called the HaPPY code. Taken from [8].

FIGURE 3.5: The HaPPY code truncated to the first three concatenation layers. The blue and red dots represent bulk and boundary degrees of freedom, respectively.

tensors in the center of the polyhedral cells, and there would be a tensor contraction through every face of the cells.

A quantity of interest for these families of architectures labeled by Schläfli symbols is the *asymptotic* code rate $R_{\{p,q\}}$. To calculate it, we can take the limit of the code rates of the truncated codes:

$$R_{\{p,q\}} = \lim_{n\to\infty} \frac{N_{\text{bulk}}}{N_{\text{boundary}}}$$

where n is the number of layers of the code. For example, as it was done in [8] for the HaPPY code, there are two types of bulk tensors per layer in this code: those that have two input legs and those that only have one input leg (shown in Figure 3.5). Denote f_n to the number of tensors in layer n that have one input leg and g_n the ones with two input legs. Taking into account how the tensors connect, we can recursively solve for f_n and g_n at each layer n:

$$\begin{pmatrix} f_n \\ g_n \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}^{n-1} \begin{pmatrix} 5 \\ 0 \end{pmatrix},$$

which gives us the following expressions:

$$f_n = 2^{-n-1}\left(\left(5-\sqrt{5}\right)\left(3+\sqrt{5}\right)^n + \left(5+\sqrt{5}\right)\left(3-\sqrt{5}\right)^n\right),$$

$$g_n = 2^{-n-1}\left(\left(3\sqrt{5}-5\right)\left(3+\sqrt{5}\right)^n - \left(5+3\sqrt{5}\right)\left(3-\sqrt{5}\right)^n\right).$$

Using these functions, we can take the limit of the code rates to obtain the asymptotic code rate of the HaPPY code

$$R_{\{4,5\}} = \lim_{n\to\infty} \frac{N_{\text{bulk}}}{N_{\text{boundary}}} = \lim_{n\to\infty} \frac{1+\sum_{k=1}^{n}(f_k+g_k)}{4f_n+3g_n} = \frac{1}{\sqrt{5}} \approx 0.447214.$$

Extending this idea of calculating code rates to other holographic codes in $\mathbb{H}^2$ labeled by Schläfi symbols by identifying different types of tensors and finding a relation between them, we get the rates for multiple values of $\{p,q\}$ shown in Table 3.1.

$\{p,q\}$	$q=4$	$q=5$	$q=6$	$q=7$	$q=8$
$p=4$	(Flat)	0.44721	0.28868	0.21822	0.17678
$p=5$	0.77123	0.39903	0.27689	0.21377	0.17469
$p=6$	0.61803	0.36603	0.26376	0.20711	0.17082

TABLE 3.1: Asymptotic code rates for different holographic codes in $\mathbb{H}^2$ labeled by Schläfi symbols.

We show the details of these calculations in Appendix B as well as pictures for the different uniform tessellations that were considered. It is worth noting that the higher code rates are achieved for smaller values of q and smaller values of p, making them possibly better candidate codes. Still, we would need to check, at least, the distance of the codes to be sure of their utility in quantum processing tasks.

3.2.2 Recovering Properties from AdS/CFT

After defining holographic states and codes, we are now interested in the holographic properties that this description of holography encapsulates and to what extent do these discrete properties hold. To begin with, as these codes have a MERA-like structure, we would expect a similar RT-formula to hold for the entropy of the tensor legs. For an arbitrary tensor network with a cut c dividing it into two tensors P and Q as shown in Figure 3.6, an explicit description of the reduced density matrix of a subsystem of uncontracted legs is obtained by tracing out the complementary subsystem.

$$|\psi\rangle = \sum_i |P_i\rangle_A \otimes |Q_i\rangle_{A^c} \implies \rho_A = \text{tr}_{A^c} |\psi\rangle\langle\psi| = \sum_{i,j} \langle Q_j \mid Q_i\rangle |P_i\rangle\langle P_j|$$

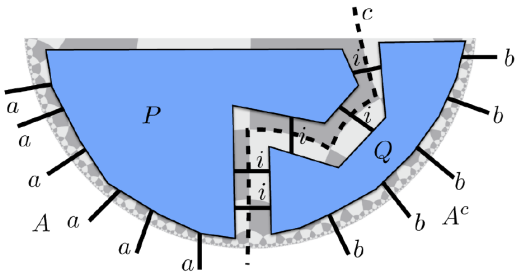


FIGURE 3.6: A tensor network divided by a cut c into two tensors P and Q. Taken from [8]

If the tensor network is the one of a holographic state, and such a network is composed of contractions of a unique perfect tensor T with all of its legs of dimension

v, like the code shown in Figure 3.4a, we can bound the entropy of ρ_A like we did in Chapter 1 for a general tensor network:

$$S(\rho_A) = -\operatorname{tr}(\rho_A \log \rho_A) \leq \min \left\{ \sum_{b \in \tilde{\gamma_A}} \log \nu \mid \tilde{\gamma_A} \text{ is a cut separating } A \text{ from } A^c \right\}.$$

Since every tensor leg contributes the same amount to the entropy, the cut that minimizes this quantity will be the *geodesic curve* γ_A defined as the cut that cuts the least number of tensor legs and separates A from A^c. So, this geometric object will bound the entropy of the subsystem:

$$S_A := S(\rho_A) \leq |\gamma_A| \log v.$$

To saturate this bound, both the tensors P and Q must be isometries from the inside to the boundary. If this were the case, it would imply that $\{|P_i\rangle\}_i$ and $\{|Q_j\rangle\}_j$ are orthogonal, achieving the equality in the entropy bound. When the tensor we are dealing with is a holographic state (over a manifold with non-positive curvature), the saturation of the bound holds [8]. This fact is not obvious, and to prove it one needs to make use of the *max-flow min-cut theorem*. The satisfied equality

$$S_A = |\gamma_A| \log(\nu)$$

is indeed a discrete analog of the RT formula (Equation 3.1). It proportionately relates the geometrical quantity $|\gamma_A|$ (the number of legs crossed by the minimal cut) with the entropy of the boundary subregion A.

On the opposite side, holographic codes do not have a direct extension of the RT formula. When cutting a holographic code using a geodesic, there will also be uncontracted tensor legs in the bulk. Then, the remaining tensors P and Q without any additional constraint will not necessarily be isometries because, despite asking holographic codes to be isometries from the bulk to the boundary, when cutting the tensor network, we are adding additional legs as an input. Thus, we have no guarantees that the remaining map will again be an isometry. The greedy algorithm is introduced to quantify the extent to which the tensors P and Q of a holographic code fail being isometries. The idea behind it is to obtain the largest possible subset network inside P or Q, which is indeed an isometry.

Definition 3.2.3. The **greedy algorithm** is a procedure that obtains a maximal cut $\gamma_A^\star$, called the **greedy geodesic**, from a boundary region A, so that the tensor between the boundary A and the cut $\gamma_A^\star$ is an isometry (from the bulk to the boundary). The steps of the algorithm are the following:

1. Set $\gamma_0 = A$, the cut that trivially cuts all tensors legs of A. Set the tensor P_0 as the identity map from A to A.

2. Identify if there is a perfect tensor T_i next to P_i such that it has half of its legs or more contracted with P_i. If not, go to step 4.

3. Set P_{i+1} as the contraction of P_i with T_i and define γ_{i+1} as the cut obtained by pushing γ_i though T_i as shown in Figure 3.7. Then, repeat step 2.

4. Return the greedy geodesic $\gamma_A^\star = \gamma_N$ where N is the last cycle of the algorithm.

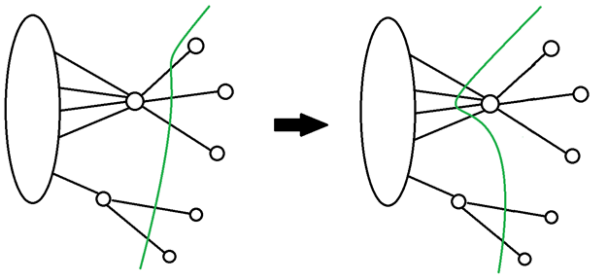


FIGURE 3.7: One iteration of the greedy algorithm passing though an admissible isometry. The green curve is the cut that will eventually converge to the greedy geodesic.

Pastawski et al. show [8] that the greedy geodesic is well defined, and the convergence of the algorithm is independent of the order in which the new perfect tensors on step 2 are chosen. This procedure allow us to quantify the extent to which the RT formula fails in holographic codes. If A is an arbitrary subregion of the boundary of a holographic state or code. Then, the entropy of A has a lower bound

$$S_A \geq |\gamma_A^\star \cap \gamma_{A^c}^\star| \log(\nu),$$

where $|\gamma_A^\star \cap \gamma_{A^c}^\star|$ represents the number of tensors legs cut by both the greedy geodesic of A and A^c. The tensor in between both cuts is called the "bipartite residual region," and when it is empty, the discrete RT equation also holds. These codes do not always saturate the bound of the RT formula, but this is no significant problem. The corrections that appear on the formula lie in the symmetries and discreteness of the codes. The importance of having a similar discrete version of the RT formula is that we are approximately recovering, from a discrete formulation, properties that hold in the field theory (continuous spaces) formulation of the AdS/CFT correspondence, making quantum error correction an impressive candidate for a discretization of this holographic correspondence.

The above description only describes a static scenario of the tensors. There are more complicated descriptions of the RT formula involving dynamic space-times. The tensor network description also appears to be adequate to describe such dynamics. May and Osborne [53, 54] include other AdS/CFT ideas to successfully add dynamics for the holographic states and codes.

Another property of the AdS/CFT correspondance that holographic states and codes also exhibit is the *causal wedge* (or AdS-Rindler) reconstruction:

Definition 3.2.4. The **causal wedge** $\mathcal{C}[A]$ of a connected boundary region A is the set of bulk tensor legs that are in between A and the greedy geodesic $\gamma_A^\star$. Likewise, the causal wedge of an arbitrary boundary region $B = \bigcup_i B_i$, where B_i is connected, is defined by $\mathcal{C}[B] := \bigcup_i \mathcal{C}[B_i]$.

By definition of the greedy algorithm, any bulk local operator $\phi(x)$ in the causal wedge of a connected boundary region A can be reconstructed as a boundary operator on A. The corresponding boundary operator that reconstructs the local operator $\phi(x)$ would be $P_{\gamma_A^\star}[\phi(x)]$, where $P_{\gamma_A^\star}$ is the isometry enclosed by the greedy geodesic $\gamma_A^\star$ and the boundary region A.

Finally, the quantum error correction perspective of AdS/CFT sheds light on the *entanglement wedge* conjecture. This conjecture assures that sometimes, you can go beyond the AdS-Rindler reconstruction and reconstruct operators that go beyond the

causal wedge of a boundary region. This idea comes from the Ryu-Takayanagi formula, where the minimal surface used to calculate the entropy of a region A in the boundary is taken over all possible surfaces with boundary equal to ∂A in the bulk. The minimal surface for a non-connected boundary region $A = \bigcup_i A_i$ is sometimes another surface different from the union of minimal surfaces for each connected component A_i. As illustrated in Figure 3.8, when the size of A is larger than half of the boundary space, the minimal surface encloses a larger region known as the **entanglement wedge** of A. In terms of holographic codes, it is defined as follows:

Definition 3.2.5. The entanglement wedge $\mathcal{E}[A]$ of a boundary region $A = \bigcup_i A_i$ is defined as the bulk region enclosed between A and the greedy geodesic $\gamma_A^\star$ obtained by applying the greedy algorithm to every connected component of the boundary region A_i at the same time.

The information of an operator living on the entanglement wedge $\mathcal{E}[A]$ but outside the causal wedge $\mathcal{C}[A]$ would be encoded, not on individual components A_i, but on the entanglement between these subregions A_i. Cotler et al. [55] proved right the entanglement wedge conjecture using universal recovery channels, which is the theory that cares about recovering information from a damaged quantum code. This shows that the quantum error correction approach for discretizing the AdS/CFT correspondence brings new tools for studying holography and provides a new theoretical formulation that strengthens its validity.

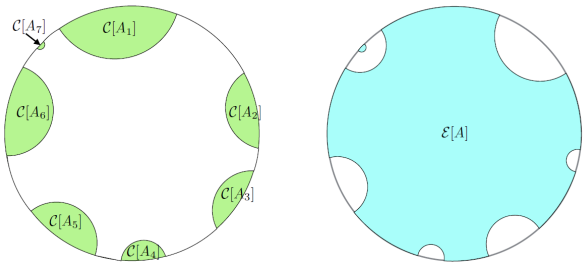


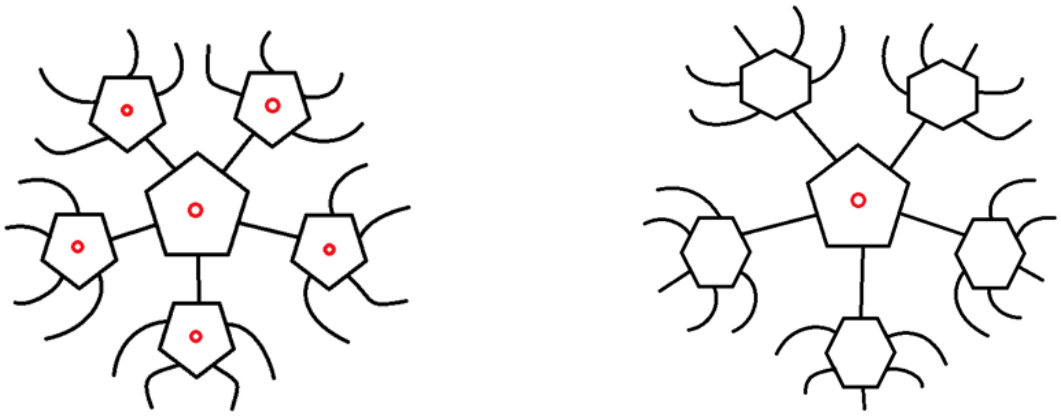
FIGURE 3.8: Causal wedge $\mathcal{C}[A]$ and entanglement wedge $\mathcal{E}[A]$ of a boundary region A. Taken from [8].

3.2.3 Holographic Codes for Quantum Error Correction

Having defined holographic quantum codes, a natural thing to ask is how good are these codes at protecting the logical qubits from errors. A naive approach would be studying the distance, error syndromes, and fidelities under noisy channels for simple holographic codes, like a truncated HaPPY code (Figure 3.4b). To do this, we use the machinery of stabilizer codes introduced in the previous chapter. Pastawski et al. show [9] the following theorem:

Theorem 3.2.1. *A holographic code that is constructed as a contraction of perfect tensors obtained from stabilizer states, and that when applying the greedy algorithm to the whole boundary of the code the greedy geodesic achieves the whole tensor network, then, the holographic code is a stabilizer code.*

This means that we can define some holographic codes with the stabilizer formalism, in particular, the stabilizer that defines the logical subspace will be a non-local stabilizer due to the geometry of these codes. Mazurek et al. [13] made a procedure to determine the stabilizer generators and logical operators of truncated holographic



(A) (B)

FIGURE 3.9: (A) The truncated HaPPY code after one concatenation layer. (B) The concatenation of the $[[5, 1, 3]]$ code with itself.

codes using the correspondence between AME states and graph states and ideas from code concatenations. In their calculations, they obtain the stabilizer matrix for the first-level truncated HaPPY code shown in Figure 3.9a. It is:

$$\begin{bmatrix}
X & Y & X & 1 & X & Y & X & 1 & Y & Z & Y & 1 & 1 & 1 & 1 & 1 & Y & Z & Y & 1 \\
Y & Z & Y & 1 & Y & Z & Y & 1 & Z & X & Z & 1 & 1 & 1 & 1 & 1 & Z & X & Z & 1 \\
Y & Z & Y & 1 & X & Y & X & 1 & X & Y & X & 1 & Y & Z & 1 & 1 & 1 & 1 & 1 & 1 \\
Z & X & Z & 1 & Y & Z & Y & 1 & Y & Z & Y & 1 & Z & X & Z & 1 & 1 & 1 & 1 & 1 \\
Z & Y & Y & Z & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
X & Z & Z & X & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & Z & Y & Y & Z & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & X & Z & Z & X & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & Z & Y & Y & Z & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & X & Z & Z & X & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & Z & Y & Y & Z & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & X & Z & Z & X & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & Z & Y & Y & Z \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & X & Z & Z & X
\end{bmatrix}.$$

If the input state of the code is a stabilizer state, the output state of the code will also be a stabilizer state for which we can calculate its stabilizer generators. In the case of $AME(6, 2)$ as an input, the stabilizer generators of the output stabilizer state will be the ones on the matrix above plus the following operators [13]:

$$\begin{bmatrix}
1 & 1 & Z & X & 1 & 1 & Z & X & X & X & X & X & X & 1 & Y & Y & X & 1 & Y & Y \\
1 & X & Z & Z & X & 1 & X & Z & X & 1 & X & Z & 1 & X & Z & Z & 1 & 1 & 1 & 1 \\
X & 1 & Y & Y & X & X & X & X & 1 & 1 & Z & X & 1 & 1 & Z & X & X & 1 & Y & Y \\
Y & Y & X & Z & Y & Z & Y & 1 & Y & Y & X & Z & Z & Z & Z & Z & Y & Z & Y & 1 \\
X & X & X & X & X & X & X & X & X & X & X & X & X & X & X & X & X & X & X & X \\
Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z
\end{bmatrix}.$$

With such a matrix, we can calculate the distance of this code and the errors associated with each syndrome. If the encoded state ρ undergoes a noisy channel $\mathcal{E}$ with some Kraus decomposition $\{E_i\}_i$ such that

$$\mathcal{E}(\rho) = p_0 I\rho I + p_1 E_1 \rho E_1^\dagger + p_2 p_1 E_2 \rho E_2^\dagger + \cdots ,$$

we can use the stabilizer generators to identify some of the errors E_i by checking the commutation relations with every element in $s_i \in \text{gen}(S)$. If the error E_i has a unique error syndrome compared to the syndromes of the other possible errors $\{E_j\}_{j\neq i}$, we will be able to correct E_i. Doing this explicitly would require an immense amount of computer memory and to overcome this we can use the binary representation of the Pauli group (Appendix C) to calculate these commutation relations. We created the *syndrome.py* module to handle syndromes of stabilizer codes and use them to obtain the distance of a code and other quantities of interest.

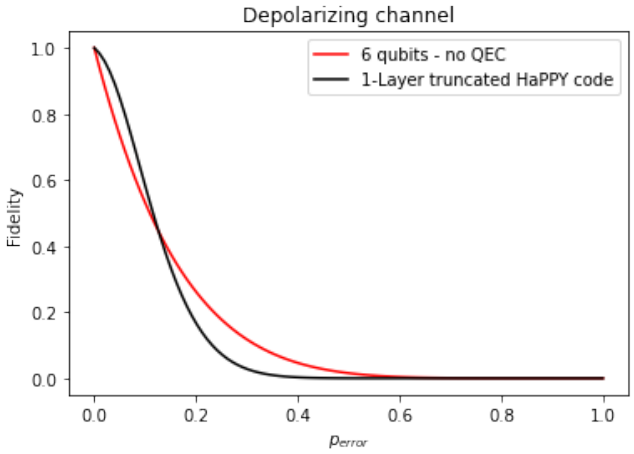
Using the python module, we found that the distance for this particular code shown in Figure 3.9a is $d = 2$. It means that if we were to use the code to protect the six logical qubits, and the quantum channel was depolarizing, we would not be able to correct every single-qubit error (which is bad). Still, comparing the fidelity of the code, assuming that we have noiseless gates to do recovery, with the fidelity of six noisy qubits, the code would slightly outperform the non-encoded state (Figure 3.10a). Nonetheless, this one-level truncated HaPPY code is not a good code, and we can get a much better encoding for six qubits. Still, it seems that the distance does not take into account the different levels of encoding, since the center qubit was encoded using a code of distance 3, and this is right, but we will mention this briefly. If we just cared about the worst-case scenario for the six qubits, the distance 2 comes from the five encoded qubits in the first concatenation layer. The outer-most tensors are $[[4, 2, 2]]$ quantum codes, since they are taking as an input two legs, making the total six-qubit AME state a distance 2 code.

We also study the code in Figure 3.9b. The stabilizer of such holographic code is easier to obtain since it is the concatenation of the $[[5, 1, 3]]$ qubit code with itself. The code shown in Figure 3.9b corresponds to a $[[25, 1, 9]]$ quantum code, since concatenating a $[[n_1, k_1, d_1]]$ code with a $[[n_2, k_2, d_2]]$ gives rise to a $[[n_1 n_2, k_1 k_2, d_1 d_2]]$ [56]. Its stabilizer is

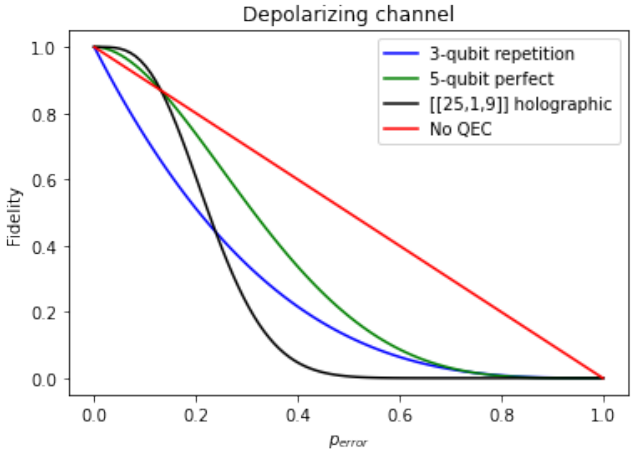
$$\begin{bmatrix}
X & Z & Z & X & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
I & X & Z & Z & X & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
X & I & X & Z & Z & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
Z & X & I & X & Z & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & X & Z & Z & X & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & I & X & Z & Z & X & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & X & I & X & Z & Z & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & Z & X & I & X & Z & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & X & Z & Z & X & I & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & I & X & Z & Z & X & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & X & I & X & Z & Z & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & Z & X & I & X & Z & I & I & I & I & I & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & X & Z & Z & X & I & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & X & Z & Z & X & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & X & I & X & Z & Z & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & Z & X & I & X & Z & I & I & I & I & I \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & X & Z & Z & X & I \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & X & Z & Z & X \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & X & I & X & Z & Z \\
I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & I & Z & X & I & X & Z \\
X & X & X & X & X & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & X & X & X & X & X & I & I & I & I & I \\
I & I & I & I & I & X & X & X & X & X & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z & X & X & X & X & X \\
X & X & X & X & X & I & I & I & I & I & X & X & X & X & X & Z & Z & Z & Z & Z & Z & Z & Z & Z & Z \\
Z & Z & Z & Z & Z & X & X & X & X & X & I & I & I & I & I & X & X & X & X & X & Z & Z & Z & Z & Z
\end{bmatrix}$$

With this stabilizer matrix, we can calculate the error syndromes, as we did with the previous code. With the error syndromes, we can obtain a lower bound for the fidelity of the encoded state, assuming we can perform a perfect decoding. We calculate this bound for the code in Figure 3.9b and compare it to familiar quantum codes, such as the 5-qubit code and the 3 qubit repetition code (Figure 3.10b).

This naive approach does not consider the geometry of the code, and we can get a better description using *operator algebra quantum error correction* [9], which



(A)



(B)

FIGURE 3.10: Lower-bound for the fidelity of a state suffering i.i.d. depolarizing errors using different quantum codes. (A) Shows the fidelity of the six-qubit state under the influence of the code of Figure 3.9a. (B) Shows the fidelity of the one-qubit state under the influence of the $[[25, 1, 9]]$ code of Figure 3.9b and other familiar one-qubit codes. It is worth noting that for both holographic codes, below some threshold error probability p_{th}, the lower-bound of the fidelity of the encoded state is better than the fidelity of the non-encoded qubits.

is yet another formulation of quantum error correction. Without entering in much detail with this formulation, the idea is that we can think of the encoded space as a collection of different Hilbert spaces, each with different levels of protection, and for each bulk leg $i \in \{1, \ldots, N_{\text{bulk}}\}$ we can associate a space $\mathcal{H}_i$ so that

$$\mathcal{H}_C = \bigotimes_{s=1}^{N_{\text{bulk}}} \mathcal{H}_s.$$

Every encoded qubit in $\mathcal{H}_s$ has a logical set of operators that can act on it, namely, $\mathcal{A}_i = B(\mathcal{H}_s)$ the set of bounded linear operators of the Hilbert space. In some sense, $\mathcal{A}_i$ is basically as the logical gates that we would apply to the encoded system $\mathcal{H}_i$. Having "dissected" the logical space in this way, we can generalize the notion of the distance of the code to the distance of a sub-algebra $\mathcal{A}_S$ where $S = \{s_1, \ldots, s_k\}$ is

a collection of labels of logical subspaces $\mathcal{H}_{s_i} \subseteq \mathcal{H}_C$. For a code $\mathcal{C}$ that encodes information in a set of B physical qubits, the distance of a sub-algebra $\mathcal{A}_s$ is defined by [9]:

$$d(\mathcal{A}_S) := \min\{|R| \mid R \subseteq B, A_S \text{ can be recovered from } R^c\},$$

where the statement "A_S can be recovered from R^c" means that if we erased R, which in terms of quantum channels is equivalent to applying

$$\mathcal{K}(\rho) := \mathrm{tr}_R(\rho),$$

we can find a recovery channel $\mathcal{R}$ such that for every operator $O \in B(\mathcal{H}_S)$ acting only on $\mathcal{H}_S$ we have that

$$\Pi_{\mathcal{C}} O \Pi_{\mathcal{C}} = \Pi_{\mathcal{C}}(R \circ \mathcal{K})^\dagger O \Pi_{\mathcal{C}}.$$

With this generalization, we can recover the definition of the distance of a code $\mathcal{C}$ we had from the previous chapter. If our sub-algebra is the total logical algebra $\mathcal{A}_L$ of the encoded system, we get that $d_{\mathcal{C}} = d(\mathcal{A}_L)$. A similar "dual" quantity, called **price**, can be defined for this code as

$$p(\mathcal{A}_S) := \min\{|R| \mid R \subseteq B, A_S \text{ can be recovered from } R\}.$$

One can show that they both satisfy the relation $d(\mathcal{A}_S) \leq p(\mathcal{A}_S)$ [9].

For a holographic code with boundary B, we can use the notion of entanglement wedge to find a more geometrical expression for the price and the distance of a sub-algebra [57]:

$$d(\mathcal{A}_S) = \min\{|A| \mid A \subseteq B, s \notin \mathcal{E}[A^c] \quad \forall s \in S\},$$
$$p(\mathcal{A}_S) = \min\{|A| \mid A \subseteq B, s \in \mathcal{E}[A] \quad \forall s \in S\}.$$

If the code $\mathcal{C}$ satisfies that for every boundary region A, $x \in \mathcal{E}[A]$ if and only if $x \notin \mathcal{E}[A^c]$, property known as *complementary recovery*, then both the price and the distance of $\mathcal{C}$ are the same

$$d(\mathcal{A}_S) = p(\mathcal{A}_S).$$

With this in mind, we get a different distance for encoded qubits deep in the bulk different from qubits near the boundary. With this perspective for distance, we get that, for the code on Figure 3.9a, the central qubit has distance $d(\mathcal{A}_{\text{center}}) = 6$ and the qubits on the first layer have distance $d(\mathcal{A}_{\text{outside}}) = 2$, recovering the naive result for the distance of the total code $d(\mathcal{A}_L) = \min_{i=1,\dots N_{\text{bulk}}} d(\mathcal{H}_i) = 2$. These different protection levels could possibly be useful for a *quantum memory*, a type of device that allows storing quantum states, such that it has an easy-to-access, but not well-protected, information on qubits encoded near the boundary, and more complicated-to-access, but well-protected qubits, close to the center of the bulk. This resembles a quantum memory with both long term (deep in the bulk) and short term (near the boundary) memories. A similar study on this topic was made by Breuckmann et al. [58], where they consider codes inspired from the hyperbolic structure of holographic codes and show that are useful for implementing a quantum memory. It would be great if this code could implement logical gates fault-tolerantly, and that would upgrade such a quantum memory to a quantum processor, but as we will mention in the following subsection, there are some complications with that.

3.3 Holographic Codes for Fault Tolerance

Fault tolerance is needed for quantum computation since the implementation of gates will always have some imperfections. Some codes, like the 7-qubit Steane code that we mentioned in the previous chapter, are good at implementing fault-tolerance gates. This code, in particular, could perform Clifford group gates transversally, and the $R_{\pi/8}$ gate was implemented via magic state injection. But not every code allows implementing gates fault-tolerantly easily, and a natural question would be if holographic codes are in any way permissible enough to implement enough logical gates to achieve universality in a fault-tolerant manner.

There is a reasonable motive for studying the applicability of holographic codes to fault tolerance, and it is the similar structure that these codes have to *topological codes*. As their name says, topological codes are quantum error-correcting codes that work by encoding information in a non-trivial topological manner. The fundamental idea behind the robustness of topological codes is *topological invariants*, which are quantities that are preserved under continuous deformations[1]. The simplest non-trivial example of a topological code is the **toric code** [2], also known as the *surface code*, and it is a quantum code over a 2D lattice wrapped around a torus. To define the code words, we begin with an $N \times N$ qubit lattice with the Hamiltonian

$$H = -\sum_v A_v - \sum_f B_f, \quad [A_v, B_f] = 0,$$

where $A_v = X_i \otimes X_j \otimes X_l \otimes X_k$ and $B_f = Z_m \otimes Z_p \otimes Z_q \otimes Z_r$. The operators A_v and B_f are known as the *vertex* and *plaquette* operators, respectively. Here, v denotes a vertex of the lattice and the indices i, j, l, and k of A_v refer to the qubits around the vertex v; also, f denotes a face of the lattice, and the indices m, p, q, and r of B_f refer to the qubits around the face f.

After this, we wrap the lattice with the shape of a torus, adding two new constraints to the ground state of the qubits system:

$$\prod_v A_v = \prod_f B_f = I.$$

The ground states of this Hamiltonian system will be the logical space of the surface quantum code. If $|\psi\rangle$ is a ground state, then

$$A_v |\psi\rangle = |\psi\rangle, \quad \forall v$$
$$B_f |\psi\rangle = |\psi\rangle, \quad \forall f.$$

This means that there are $2(L^2-1)$ independent constraints over $|\psi\rangle$, which make up a total of $\frac{2^N}{2^{L^2-1}2^{L^2-1}} = \frac{2^N}{2^{N-2}} = 4$ possible ground states (equivalent to two qubits). In this formalism, qubits are represented by strings of operators around one of the two holes of the torus. Due to the definition of the vertex and plaquette operators, homotopic strings of operators are equivalent.

Following a similar idea, we can define larger topological codes for dimension $D \geq 3$. These codes can do better at implementing gates which are robust to noise. Some topological codes can fault-tolerantly implement a wider set of gates, as compared to the Clifford group. This larger set of gates contain the Clifford group and is known

[1] More exactly, quantities preserved under homeomorphisms.

as the **Clifford hierarchy**. This hierarchy is defined inductively as:

$$\mathcal{C}^{(2)} := \left\{ T \in U\left(2^n\right) \mid T\mathcal{P}_n T^\dagger = \mathcal{P}_n \right\},$$
$$\mathcal{C}^{(k)} := \left\{ T \in U\left(2^n\right) \mid T\mathcal{P}_n T^\dagger \subseteq \mathcal{C}^{(k-1)} \right\}.$$

The set $\mathcal{C}^{(2)}$ is the Clifford group, and $\mathcal{C}^{(k)}$ is called the k-th level of the Clifford hierarchy. Depending on the dimension D, some topological codes can achieve implementing $C^{(D)}$ gates, making them great candidates for real-life quantum error correcting codes. One would argue that the structure of holographic codes resembles that of topological codes and thus expect the ability to implement fault-tolerantly gates beyond the Clifford group.

In a recent work [57], Cree et al. study the validity of the previous statement, and found that the set of implementable transversal gates by hyperbolic holographic codes restricts to the Clifford group $\mathcal{C}^{(2)}$, no matter the bulk and boundary dimensions. The argument to show such a strong restriction for transversal gates resides on a work by Pastawski et al. [59] that shows that for a (bulk) subsystem A of a stabilizer code, if the physical space can be partitioned into three regions $\{R_1, R_2, (R_1 \cup R_2)^c\}$ with the property that A can be recovered when erasing anyone of the three regions, then, the implementable transversal gates by the stabilizer code are restricted to the Clifford group gates.

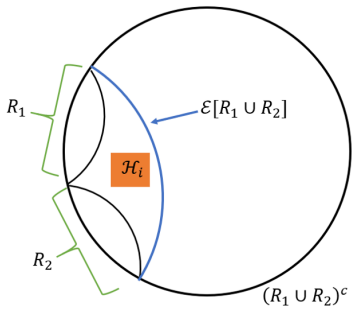


FIGURE 3.11: An example of three regions that partition the physical space and for which any two regions allow reconstructing the logical qubits in $\mathcal{H}_i$.

The work by Cree et al. [57] focuses on showing that for hyperbolic holographic codes, given an operator applied in a bulk leg, one can find three disjoint regions for which the operator can be recovered on any two of the three boundary regions as shows Figure 3.11. Naively, to find the physical implementation of a gate U on a logical qubit $|\psi_i\rangle$, we would need to push the operator U through the tensor network using the isometry defined by the holographic code itself. The physical qubits to which U can be pushed is wide enough to partition the boundary in three regions such that the property mention before holds.

This is not the end for using holographic codes in fault-tolerant protocols; as we saw before, we can implement non-Clifford gates using magic state injection protocols, but this will require some more work. Holographic codes have multiple advantages regarding the protection of information, but perhaps, this massive protection is what makes handling information fault-tolerantly much harder. This is something that could be studied in the future, bringing new hope for these possibly called *holographic fault-tolerant quantum computers*.

Chapter 4

Perspectives on Holographic Codes

We have presented an introduction to a recently discovered family of quantum error-correcting codes called *holographic codes* [9]. These codes are a contraction of many perfect tensors that induce an isometry from the input space (bulk) to the output space (boundary). Holographic codes were created for one purpose, explaining, from a quantum information point of view, the AdS/CFT correspondence [4]. They are a proposal to discretize the AdS/CFT correspondence. In the limiting scenario, where every encoded qubit plays the role of a point on an AdS space, they reproduce many of the expected properties from holography. We should ask: why are holographic codes relevant?

- First, these codes help understanding why the Ryu-Takayanagi formula is valid. This formula relates the entropy S_A, an information-theoretic variable, of a region A of the CFT, with a geodesic γ_A, a geometric quantity, in the bulk. In the discrete version of holographic states (with tensor bonds of dimension ν), a similar equation is valid:

$$\begin{aligned} \textbf{AdS/CFT (Continuous):} \quad & S_A = |\gamma_A|(4G)^{-1}, \\ \textbf{QEC (Discrete):} \quad & S_A = |\gamma_A| \log(\nu). \end{aligned}$$

  Similarly, this discrete proposal provides a new way to understand the AdS-Rindler reconstruction from a quantum error correction point of view [4]. The boundary reconstruction of an operator $\mathcal{O}(x)$ from the bulk that lives in the causal wedge of a region A can be interpreted as the error correction recovery of $\mathcal{O}(x)$ after the quantum erasure channel

$$\mathcal{K}(\rho) := \mathrm{tr}_{A^c}(\rho)$$

  deletes the physical qubits of A^c. This perspective explains, not only the reconstruction property of the AdS-Rindler reconstruction, but also its non-uniqueness. This is because quantum error-correcting codes encode information redundantly so that if some component fails, the information can be recovered from the non-disturbed components. Holographic codes provide a completely rigorous framework to understand holography from a quantum mechanical point of view. Studying them further could help us explain more phenomena present in AdS/CFT, such as black holes, corresponding to high-energy states in the boundary theory, as well as the description of the dynamics for different field operators.

- Second, holographic codes are, at their core, quantum error-correcting codes. From a quantum computing perspective, these codes have finite code rates, and if created correctly, they could have high code distances for protecting the

encoded words. As discussed in this work, holographic quantum codes could be used for implementing a quantum memory with different levels of protection that could be exploited for short-term and long-term storing of quantum states. This is because the distance of the sub-algebras depends on how deep the considered bulk point is from the boundary [9, 57]:

$$d(\mathcal{A}_S) = \min\{|A| \mid A \subseteq B, s \notin \mathcal{E}[A^c] \quad \forall s \in S\}\,.$$

The effect of a depolarizing channel acting over small-sized holographic codes (Figures 3.10b and 3.10a) show that these codes help to reduce the error probability in the idealized scenario of non-faulty gates. When considering real-life applications, these codes should allow the implementation of fault-tolerant gates to be considered as potential code candidates for quantum computing. Recently, Cree et al. [57] showed that the set of implementable transversal gates in holographic codes are restricted to Clifford group gates, bringing up new challenges to the implementation of fault-tolerant protocols involving holographic codes.

Finally, holographic codes and states by themselves are a tool that could be further exploited in other fields. For example, in condensed matter physics, for systems that exhibit holographic properties, or in quantum cryptography, for finding protocols that secure information against adversarial parties in a way that less-important data is encoded near the boundary and more-important data deep in the bulk. Ultimately, the discovery of holographic codes could lead, in a future where fault-tolerant quantum computing is a reality, to simulate toy models of AdS spaces and discover unimaginable facts about it, like a better understanding of black holes via a simulation of a CFT with a corresponding holographic dual, without the need of ever getting close to one. It is only a matter of time for humanity to keep pushing our knowledge boundaries and continue discovering the unknown.

Appendix A

Classical Error Correction

Classical error correction is a field of information theory concerned with protecting classical data, i.e. bits. The idea is to encode messages written with a set of letters so that they become robust against possible errors. For this thesis, it will be enough to define linear codes as they are analogous to stabilizer codes in quantum error correction.

Linear Codes

To define a linear code, we must remember what a *field* is:

Definition A.0.1. A **field** F is a set of elements together with two operations, multiplication $\cdot : F \times F \to F$ and addition $+ : F \times F \to F$, such that the following properties are satisfied for every $a, b, c \in F$:

- Associativity: $a + (b + c) = (a + b) + c$ and $a \cdot (b \cdot c) = (a \cdot b) \cdot c$.
- Commutativity: $a + b = b + a$ and $a \cdot b = b \cdot a$.
- Identity: there exists two elements 0 and 1 in F such that $a+0 = a$ and $a \cdot 1 = a$.
- Distributivity: $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$.
- Inverse: For $a \neq 0$, there exists two elements $-a$ and a^{-1} in F such that $a + (-a) = 0$ and $a \cdot a^{-1} = 1$.

If $|F| < \infty$, then F is isomorphic to some $\mathbb{F}_n := \mathbb{Z}/n\mathbb{Z}$. This definition will help us to formalize what we mean by a set of letters, and so, a message will be a sequences of elements of a field F.

Definition A.0.2. A **linear code** that encodes k letters in n letters is a is a linear subspace $\mathcal{C} \subset \mathbb{F}_q^n$ of dimension k. Since $\mathcal{C}$ is a linear subspace, it is generated by a basis of size k, and using this basis we can define the **generating matrix** G of $\mathcal{C}$ as the $k \times (n - k)$ matrix with the rows being the generating basis. With G we can obtain a direct encoding of the words $\mathbb{F}_q^k$ as $\mathcal{C} = \mathbb{F}_q^n[I_{k\times k}|G]$.

To define a metric for this space, this is, a way of measuring distance between words, we use the **Hamming distance**. It is defined as the number of positions where two words differ. For example, the Hamming distance between 0011 and 0101 is 2. The **distance** of a code $\mathcal{C}$ is the minimum Hamming distance between two codewords $a, b \in \mathcal{C}$. Linear codes of length n, dimension k, and distance d are denoted by $[n, k, d]$[1].

[1]Fun fact: this is why quantum stabilizer codes are denoted by $[[n, k, d]]$

Another equivalent way of defining the code $\mathcal{C}$ is using an orthogonal basis that generates the orthogonal complement of the code $\mathcal{C}^{\perp}$. Using the orthogonal basis we can create a $(n-k)\times n$ matrix called the **parity check matrix** H of $\mathcal{C}$. It defines an orthogonal code called *dual* code $\mathcal{C}^{\perp} := H\mathbb{F}_q^n$, and it satisfies that $H(G^T\mathbb{F}_q^k) = 0$. Its name comes from checking the parity of the code and detecting errors. If $v \in \mathbb{F}_q^k$ is a message, and $e \in \mathbb{F}_q^n$ is an error, then

$$H(G^T v + e) = H(G^T v) + H(e) = H(e).$$

The vector $H(e)$ is known as the error syndrome of e because it helps to detect if an error has affected the original message or not.

An important result of Classical coding theory is the Singleton bound. For a linear code $[n, k, d]$, this bound relates these three values as

$$d \leq n - k + 1.$$

Codes that saturate this bound are called **maximum-distance separable** (MDS) codes. MDS codes are relevant for this thesis since they were used in Chapter 2 for constructing AME states.

Appendix B

Asymptotic Code Rates

Calculating different Code Rates

This appendix will show how to calculate the code rates for the different codes in the Table 3.1. I follow the idea that was used by Pastawski et al [8]. The calculation is done by counting each tensor of the tensor network from the inside out. There are multiple types of tensors and are classified depending on the number and structure of input and output legs. Throughout this appendix, the number of different types of tensors per layer k are identified with f_k, g_k, h_k, and l_k.

- **Schläfli symbol $\{4,\cdot\}$ (Figure B.1)**

(A) $\{4,5\}$**:**

$$\begin{pmatrix} f_n \\ g_n \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}^{n-1} \begin{pmatrix} 5 \\ 0 \end{pmatrix}$$

$$f_n = 2^{-n-1}\left(\left(5-\sqrt{5}\right)\left(3+\sqrt{5}\right)^n + \left(5+\sqrt{5}\right)\left(3-\sqrt{5}\right)^n\right)$$

$$g_n = 2^{-n-1}\left(\left(3\sqrt{5}-5\right)\left(3+\sqrt{5}\right)^n - \left(5+3\sqrt{5}\right)\left(3-\sqrt{5}\right)^n\right)$$

$$R_{\{4,5\}} = \lim_{n\to\infty} \frac{N_{\text{bulk}}}{N_{\text{boundary}}} = \lim_{n\to\infty} \frac{1+\sum_{k=1}^{n}(f_k+g_k)}{4f_n+3g_n} = \frac{1}{\sqrt{5}} \approx 0.447214$$

(B) $\{4,6\}$

$$\begin{pmatrix} f_n \\ g_n \end{pmatrix} = \begin{pmatrix} 3 & 2 \\ 1 & 1 \end{pmatrix}^{n-1} \begin{pmatrix} 6 \\ 0 \end{pmatrix}$$

$$R_{\{4,6\}} = \lim_{n\to\infty} \frac{N_{\text{bulk}}}{N_{\text{boundary}}} = \lim_{n\to\infty} \frac{1+\sum_{k=1}^{n}(f_k+g_k)}{5f_n+4g_n} = \frac{1}{2\sqrt{3}} \approx 0.288675$$

(C) $\{4,7\}$

$$\begin{pmatrix} f_n \\ g_n \end{pmatrix} = \begin{pmatrix} 4 & 3 \\ 1 & 1 \end{pmatrix}^{n-1} \begin{pmatrix} 7 \\ 0 \end{pmatrix}$$

$$R_{\{4,7\}} = \lim_{n\to\infty} \frac{N_{\text{bulk}}}{N_{\text{boundary}}} = \lim_{n\to\infty} \frac{1+\sum_{k=1}^{n}(f_k+g_k)}{6f_n+5g_n} = \frac{1}{\sqrt{21}} \approx 0.218218$$

(D) $\{4, 8\}$

$$\begin{pmatrix} f_n \\ g_n \end{pmatrix} = \begin{pmatrix} 5 & 4 \\ 1 & 1 \end{pmatrix}^{n-1} \begin{pmatrix} 8 \\ 0 \end{pmatrix}$$

$$R_{\{4,8\}} = \lim_{n\to\infty} \frac{N_{\text{bulk}}}{N_{\text{boundary}}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k)}{7f_n + 6g_n} = \frac{1}{4\sqrt{2}} \approx 0.176777$$

- **Schläfli symbol $\{5, \cdot\}$ (Figure B.2)**

(A) $\{5, 4\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \\ l_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 2 & 1 & 1 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 4 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{5,4\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k + l_k)}{3f_n + 2g_n + 2.5h_n + 2l_n} \approx 0.77123$$

(B) $\{5, 5\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \\ l_n \end{pmatrix} = \begin{pmatrix} 2 & 1 & 2 & 1 \\ 2 & 1 & 1 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 5 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{5,5\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k + l_k)}{4f_n + 3g_n + 3.5h_n + 3l_n} \approx 0.399031$$

(C) $\{5, 6\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \\ l_n \end{pmatrix} = \begin{pmatrix} 3 & 2 & 3 & 2 \\ 2 & 1 & 1 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 6 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{5,6\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k + l_k)}{5f_n + 4g_n + 4.5h_n + 4l_n} \approx 0.276887$$

(D) $\{5, 7\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \\ l_n \end{pmatrix} = \begin{pmatrix} 4 & 3 & 4 & 3 \\ 2 & 1 & 1 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 7 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{5,7\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k + l_k)}{6f_n + 5g_n + 5.5h_n + 5l_n} \approx 0.213767$$

(E) $\{5,8\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \\ l_n \end{pmatrix} = \begin{pmatrix} 5 & 4 & 5 & 4 \\ 2 & 1 & 1 & 2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 8 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{5,8\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k + l_k)}{7f_n + 6g_n + 6.5h_n + 6l_n} \approx 0.174695$$

- **Schläfli symbol** $\{6,\cdot\}$ **(Figure B.3)**

(A) $\{6,4\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 2 & 1 & 2 \\ 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 4 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{6,4\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k)}{3f_n + 2.5g_n + 2h_n} \approx 0.618034$$

(B) $\{6,5\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \end{pmatrix} = \begin{pmatrix} 2 & 2 & 1 \\ 2 & 1 & 2 \\ 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 5 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{6,5\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k)}{4f_n + 3.5g_n + 3h_n} \approx 0.366025$$

(C) $\{6,6\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \end{pmatrix} = \begin{pmatrix} 3 & 3 & 2 \\ 2 & 1 & 2 \\ 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 6 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{6,6\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k)}{5f_n + 4.5g_n + 4h_n} \approx 0.263763$$

(D) $\{6,7\}$**:**

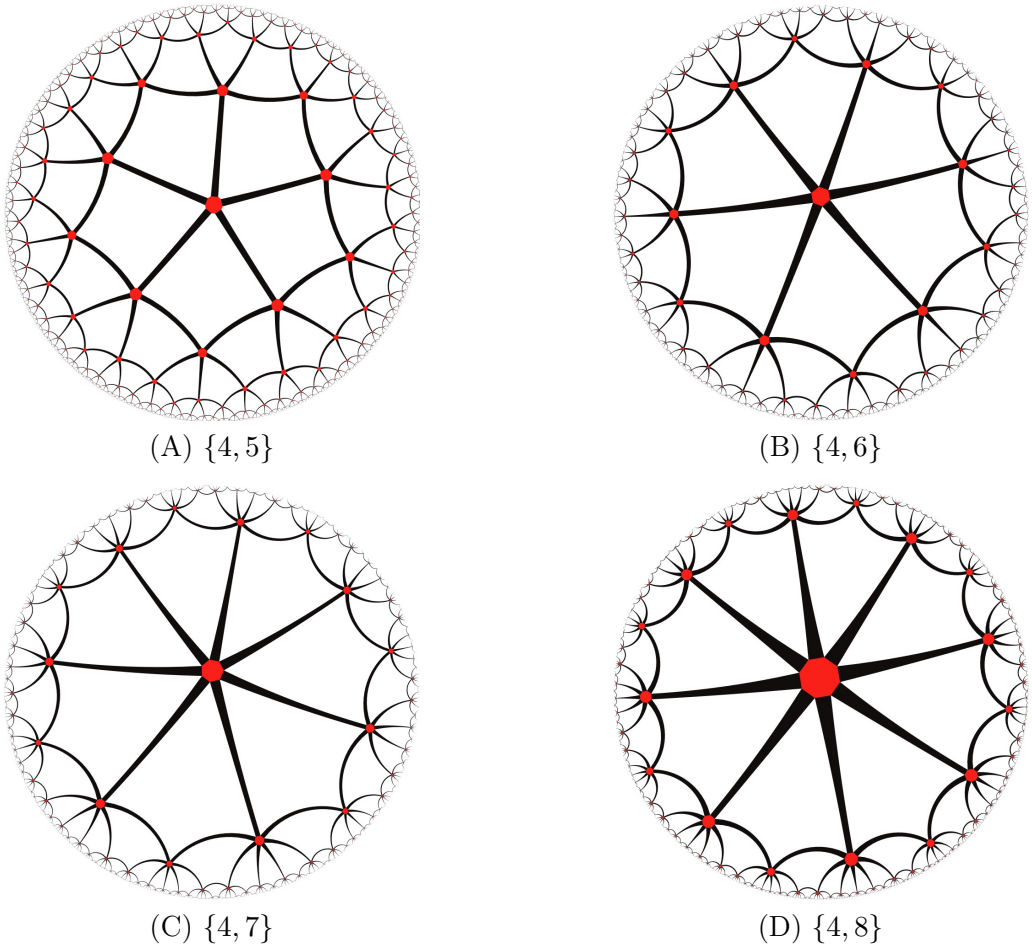
$$\begin{pmatrix} f_n \\ g_n \\ h_n \end{pmatrix} = \begin{pmatrix} 4 & 4 & 3 \\ 2 & 1 & 2 \\ 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 7 \\ 0 \\ 0 \end{pmatrix}$$

$$R_{\{6,7\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k)}{6f_n + 5.5g_n + 5h_n} \approx 0.207107$$

(E) $\{6,8\}$**:**

$$\begin{pmatrix} f_n \\ g_n \\ h_n \end{pmatrix} = \begin{pmatrix} 5 & 5 & 4 \\ 2 & 1 & 2 \\ 0 & \frac{1}{2} & 0 \end{pmatrix}^{n-1} \begin{pmatrix} 8 \\ 0 \\ 0 \end{pmatrix}$$

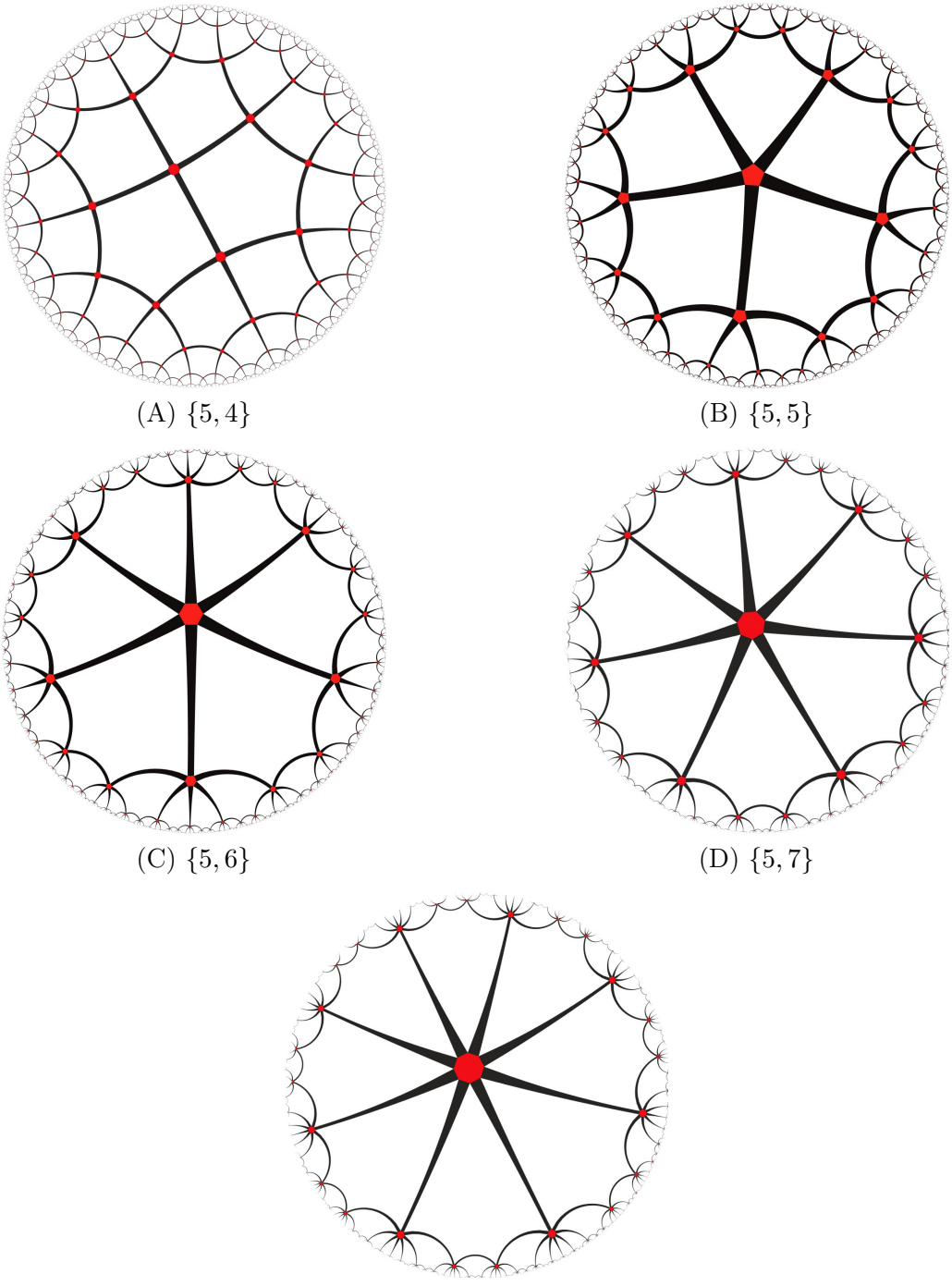
$$R_{\{6,8\}} = \lim_{n\to\infty} \frac{1 + \sum_{k=1}^{n} (f_k + g_k + h_k)}{7f_n + 6.5g_n + 6h_n} \approx 0.17082$$



(A) $\{4, 5\}$ (B) $\{4, 6\}$

(C) $\{4, 7\}$ (D) $\{4, 8\}$

FIGURE B.1: Different holographic codes with Schläfli symbol $\{4, \cdot\}$. Created with KaleidoTile [60].

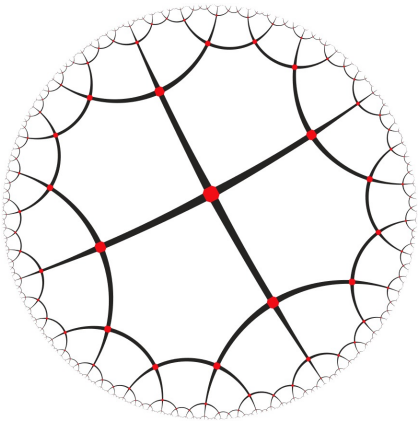


(A) $\{5, 4\}$ (B) $\{5, 5\}$

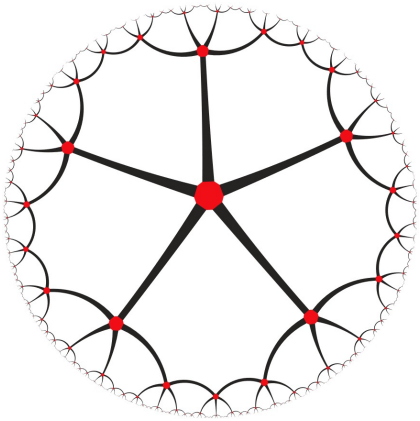
(C) $\{5, 6\}$ (D) $\{5, 7\}$

(E) $\{5, 8\}$

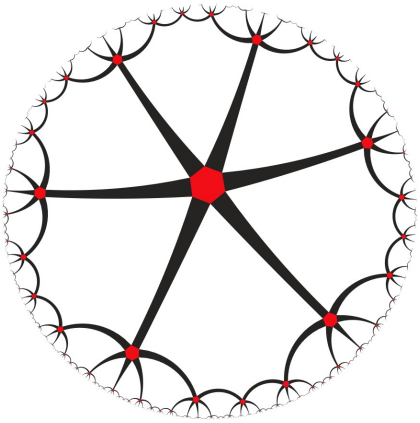
FIGURE B.2: Different holographic codes with Schläfli symbol $\{5, \cdot\}$. Created with KaleidoTile [60].



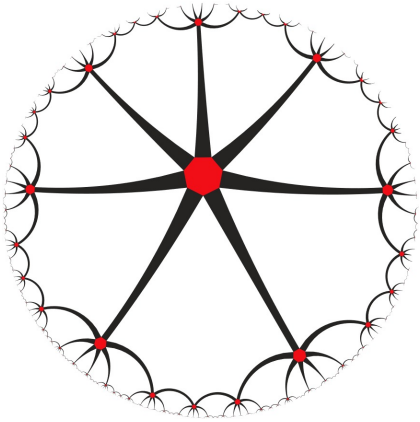
(A) $\{6, 4\}$



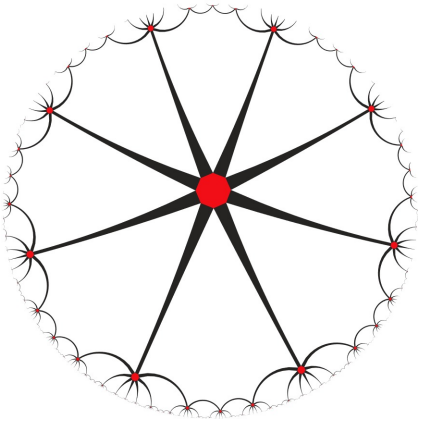
(B) $\{6, 5\}$



(C) $\{6, 6\}$



(D) $\{6, 7\}$



(E) $\{6, 8\}$

FIGURE B.3: Different holographic codes with Schläfli symbol $\{6, \cdot\}$. Created with KaleidoTile [60].

Appendix C

Syndrome.py

For some calculations of this thesis, we used some code that can be found on this link. The main idea behind the code is explained in this appendix.

Binary representation of Pauli group

We can express the Pauli group elements as binary strings. For Stabilizer codes, this simplifies calculations because we do not need to multiply the exponentially large Pauli group elements but only work with a linear number of terms with the binary strings. Each Pauli group element has a corresponding binary vector. For example, for the $[[5, 1, 3]]$ qubit code whose stabilizer is

$$\begin{aligned} &X \otimes Z \otimes Z \otimes X \otimes 1 \\ &1 \otimes X \otimes Z \otimes Z \otimes X \\ &X \otimes 1 \otimes X \otimes Z \otimes Z \\ &Z \otimes X \otimes 1 \otimes X \otimes Z \end{aligned}$$

its binary representation is

$$\left(\begin{array}{ccccc|ccccc} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right)$$

Here, every X operator is a 1 on the left matrix, and every Z operator is a 1 on the right one. Since Y operators are a product of X and Z, it corresponds to both a 1 in the left and the right matrix.

In this binary vector correspondence, the commutation relation between two operators A and B is translated into a symplectic product between their corresponding binary representation. If A and B are represented by the binary vectors $(a|b)$ and $(c|d)$ respectively, the relation $[A, B] = 0$ would correspond to

$$Q(a|b, c|d) = \sum_{i=1}^{n} (a_i d_i + b_i c_i) = 0.$$

We used this binary representation of the Pauli group to calculate the error syndromes for a particular stabilizer. The file syndrome.py contains multiple functions to simplify these calculations. Additionally, the Stabilizer.exe file is a simple GUI for some of the functions of syndrome.py. It is the same application as the file GUI.py with the only difference that it does not require to be run on the console.

Bibliography

[1] D. Gottesman, arXiv:0904.2557 [quant-ph] (2009).

[2] A. Kitaev, Annals of Physics **303**, 2 (2003).

[3] Y.-C. Lee, C. Brell, and S. T. Flammia, Journal of Statistical Mechanics: Theory and Experiment **2017**, 083106 (2017).

[4] A. Almheiri, X. Dong, and D. Harlow, Journal of High Energy Physics **2015**, 163 (2015).

[5] L. Susskind, Journal of Mathematical Physics **36**, 6377 (1995).

[6] J. M. Maldacena, International Journal of Theoretical Physics **38**, 1113 (1999).

[7] A. Hamilton, D. Kabat, G. Lifschytz, and D. A. Lowe, Physical Review D **73**, 086003 (2006).

[8] F. Pastawski, B. Yoshida, D. Harlow, and J. Preskill, Journal of High Energy Physics **2015**, 149 (2015).

[9] F. Pastawski and J. Preskill, Physical Review X **7**, 021022 (2017).

[10] W. Helwig, W. Cui, J. I. Latorre, A. Riera, and H.-K. Lo, Physical Review A **86**, 052335 (2012).

[11] W. Helwig and W. Cui, arXiv:1306.2536 [quant-ph] (2013).

[12] D. A. Lidar and T. A. Brun, *Quantum error correction*, Cambridge university press, 2013.

[13] P. Mazurek, M. Farkas, A. Grudka, M. Horodecki, and M. Studziński, Physical Review A **101**, 042305 (2020).

[14] A. F. Rubio, Holographic quantum error correcting codes, 2016.

[15] J. Biamonte, arXiv:1912.10049 [cond-mat, physics:math-ph, physics:quant-ph] (2020).

[16] W. Helwig, arXiv:1306.2879 [quant-ph] (2013).

[17] S. Ryu and T. Takayanagi, Journal of High Energy Physics **2006**, 045 (2006).

[18] M. A. Nielsen and I. Chuang, *Quantum computation and quantum information*, American Association of Physics Teachers, 2002.

[19] R. Horodecki, P. Horodecki, M. Horodecki, and K. Horodecki, Reviews of Modern Physics **81**, 865 (2009).

[20] S. Lloyd, Science **273**, 1073 (1996).

[21] G. Vidal, Journal of Modern Optics **47**, 355 (2000), arXiv: quant-ph/9807077.

[22] M. A. Nielsen, Lecture Notes, Department of Physics, University of Queensland, Australia (2002).

[23] W. F. Stinespring, Proceedings of the American Mathematical Society **6**, 211 (1955).

[24] C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. D. Sarma, Reviews of Modern Physics **80**, 1083 (2008).

[25] S. Bartolucci et al., arXiv preprint arXiv:2101.09310 (2021).

[26] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, arXiv preprint quant-ph/0001106 (2000).

[27] R. Raussendorf and H. J. Briegel, Physical Review Letters **86**, 5188 (2001).

[28] A. Jahn and J. Eisert, arXiv:2102.02619 [cond-mat, physics:hep-th, physics:quant-ph] (2021), arXiv: 2102.02619.

[29] J. Eisert, arXiv preprint arXiv:1308.3318 (2013).

[30] G. Evenbly and G. Vidal, Journal of Statistical Physics **145**, 891 (2011), arXiv: 1106.1082.

[31] D. Gottesman, Quantum Error Correction, https://www.perimeterinstitute.ca/personal/dgottesman/QECC2018/index.html.

[32] A. M. Steane, A Tutorial on Quantum Error Correction.

[33] J. Preskill, Lecture notes on quantum computation, http://theory.caltech.edu/∼preskill/ph229/.

[34] D. J. MacKay and D. J. Mac Kay, *Information theory, inference and learning algorithms*, Cambridge university press, 2003.

[35] S. A. Aly, arXiv:0711.4603 [quant-ph] (2007), arXiv: 0711.4603.

[36] P. W. Shor, Fault-tolerant quantum computation, in *Proceedings of 37th Conference on Foundations of Computer Science*, pages 56–65, IEEE, 1996.

[37] G. Nebe, E. M. Rains, and N. J. Sloane, Designs, Codes and Cryptography **24**, 99 (2001).

[38] D. Gottesman, arXiv preprint quant-ph/9807006 (1998).

[39] S. Aaronson and D. Gottesman, Physical Review A , 14 (2004).

[40] X. Chen, H. Chung, A. W. Cross, B. Zeng, and I. L. Chuang, Physical Review A **78**, 012353 (2008), arXiv: 0801.2360.

[41] D. Gottesman and I. L. Chuang, Nature **402**, 390 (1999).

[42] F. Huber, C. Eltschka, J. Siewert, and O. Gühne, Journal of Physics A: Mathematical and Theoretical **51**, 175301 (2018), arXiv: 1708.06298.

[43] F. Huber, O. Gühne, and J. Siewert, Physical Review Letters **118**, 200502 (2017).

[44] F. Huber and N. Wyderka, Table of ame states, 2019.

[45] A. Cervera-Lierta, J. I. Latorre, and D. Goyeneche, Physical Review A **100**, 022342 (2019), arXiv: 1904.07955.

[46] M. Bahramgiri and S. Beigi, arXiv:quant-ph/0610267 (2007), arXiv: quant-ph/0610267.

[47] J. D. Bekenstein, Physical Review D **23**, 287 (1981).

[48] S. Ryu and T. Takayanagi, Phys. Rev. Lett. **96**, 181602 (2006).

[49] A. Hamilton, D. Kabat, G. Lifschytz, and D. A. Lowe, Physical Review D **74**, 066009 (2006), arXiv: hep-th/0606141.

[50] B. Swingle, arXiv:1209.3304 [cond-mat, physics:hep-th, physics:quant-ph] (2012), arXiv: 1209.3304.

[51] B. Swingle, Physical Review D **86**, 065007 (2012).

[52] J. I. Latorre and G. Sierra, arXiv:1502.06618 [cond-mat, physics:hep-th, physics:quant-ph] (2015), arXiv: 1502.06618.

[53] A. May, Journal of High Energy Physics **2017**, 118 (2017).

[54] T. J. Osborne and D. E. Stiegemann, arXiv:1706.08823 [hep-th, physics:math-ph, physics:quant-ph] (2020), arXiv: 1706.08823.

[55] J. Cotler et al., Physical Review X **9**, 031011 (2019).

[56] D. Gottesman, arXiv preprint quant-ph/9705052 (1997).

[57] S. Cree, K. Dolev, V. Calvera, and D. J. Williamson, arXiv:2103.13404 [hep-th, physics:quant-ph] (2021), arXiv: 2103.13404.

[58] N. P. Breuckmann, C. Vuillot, E. Campbell, A. Krishna, and B. M. Terhal, Quantum Science and Technology **2**, 035007 (2017).

[59] F. Pastawski and B. Yoshida, Physical Review A **91**, 012305 (2015).

[60] J. Weeks, A computer program for creating spherical, Euclidean and hyperbolic tilings (2020).